

SECRET

FIG. 1

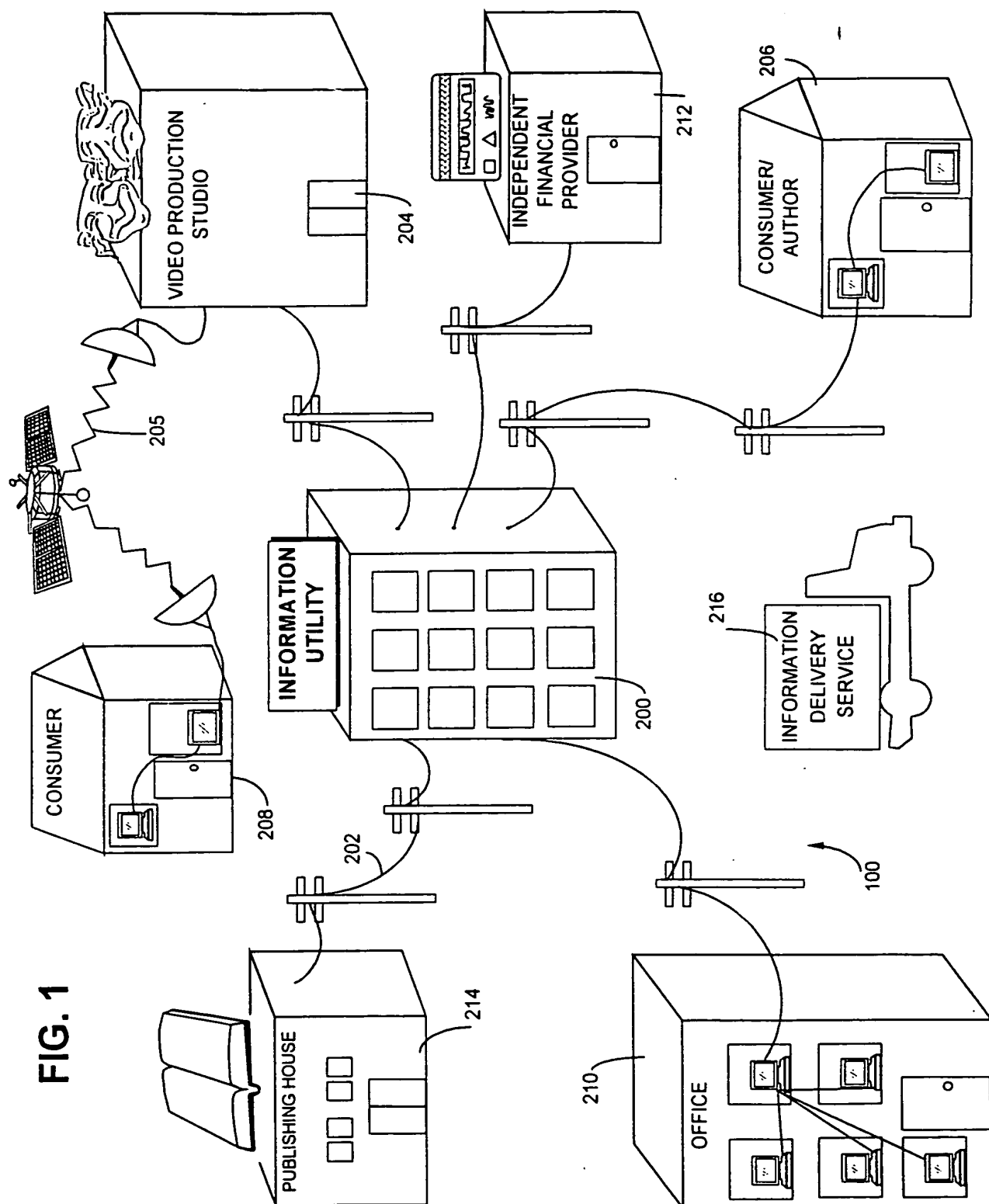


FIG. 1A

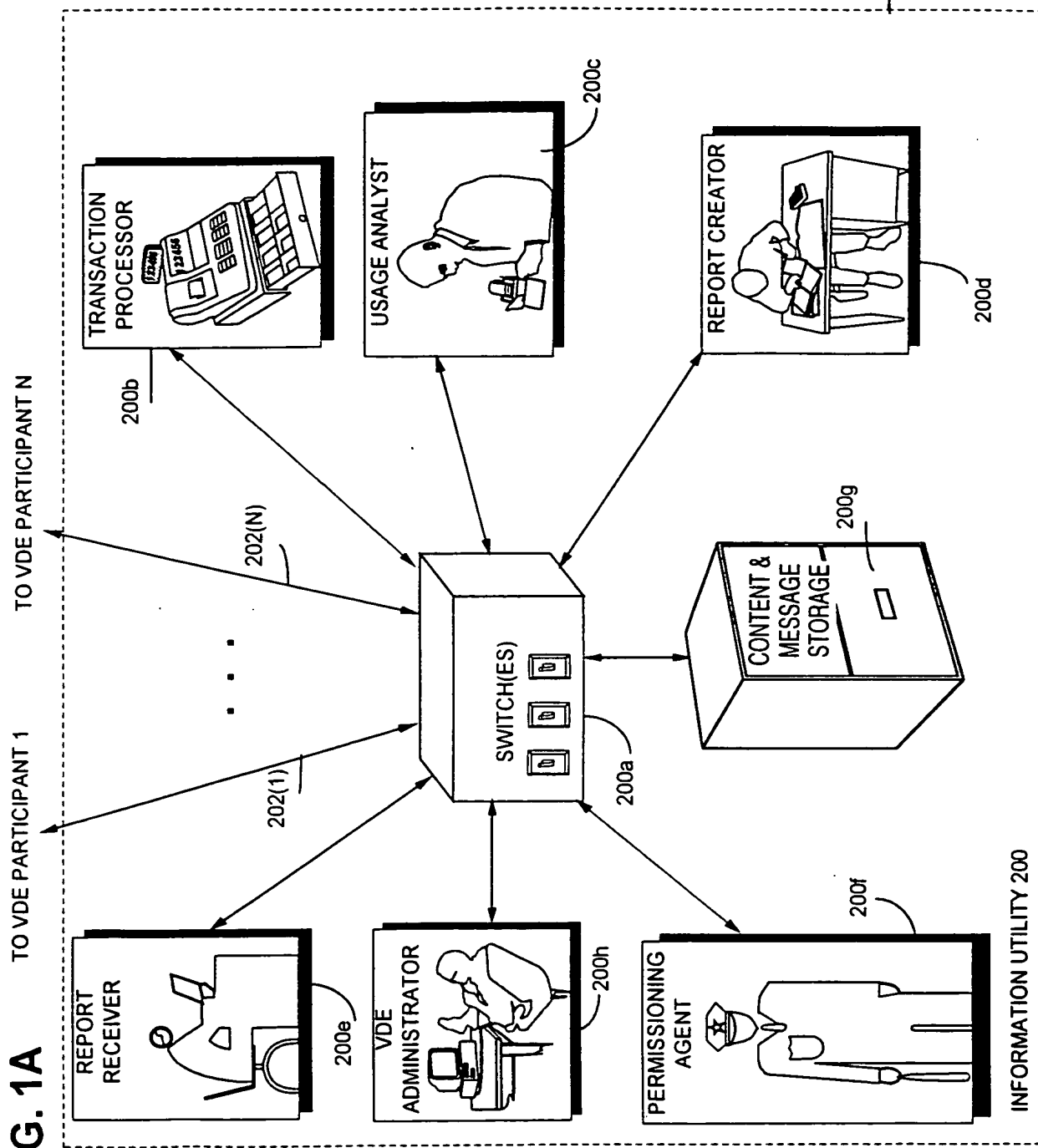


FIG. 2

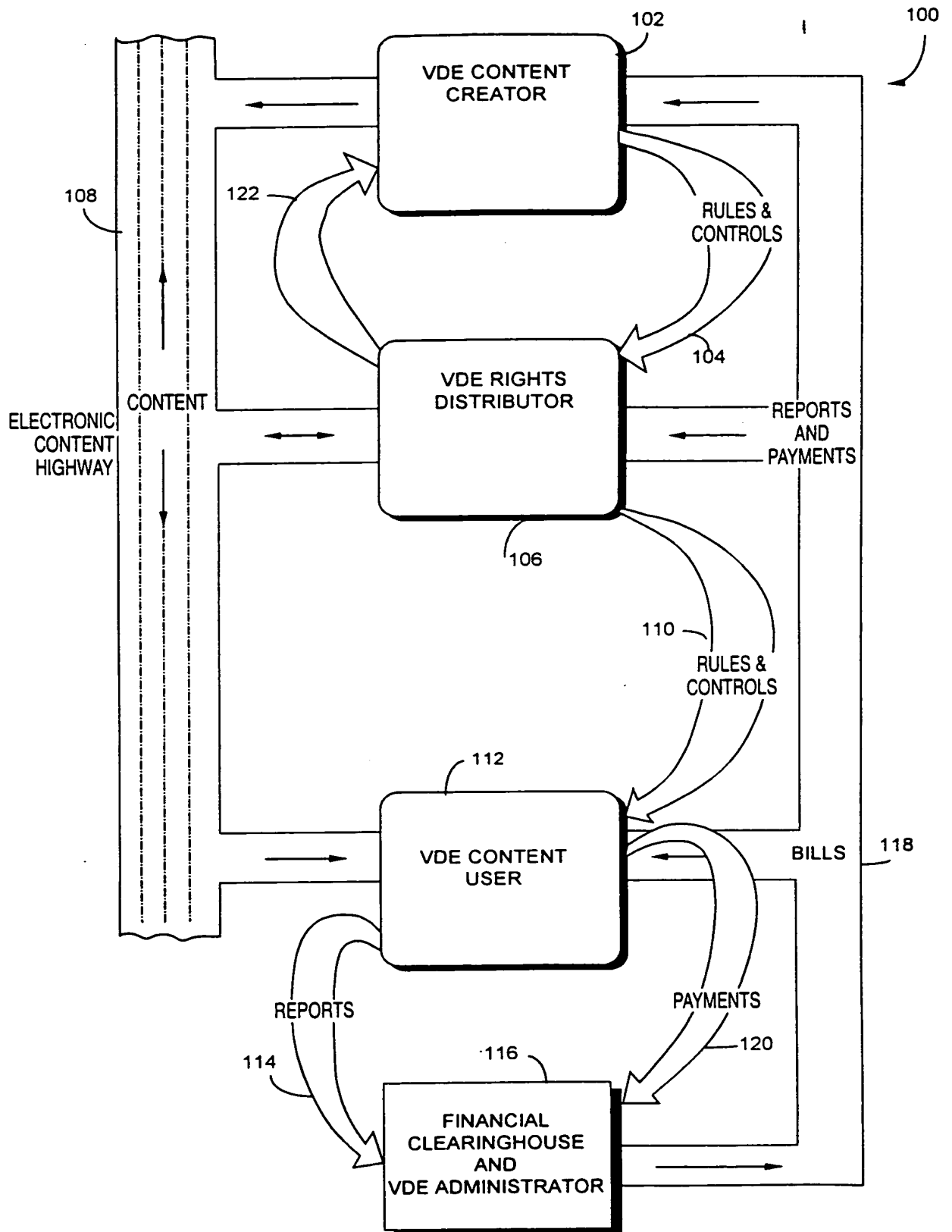
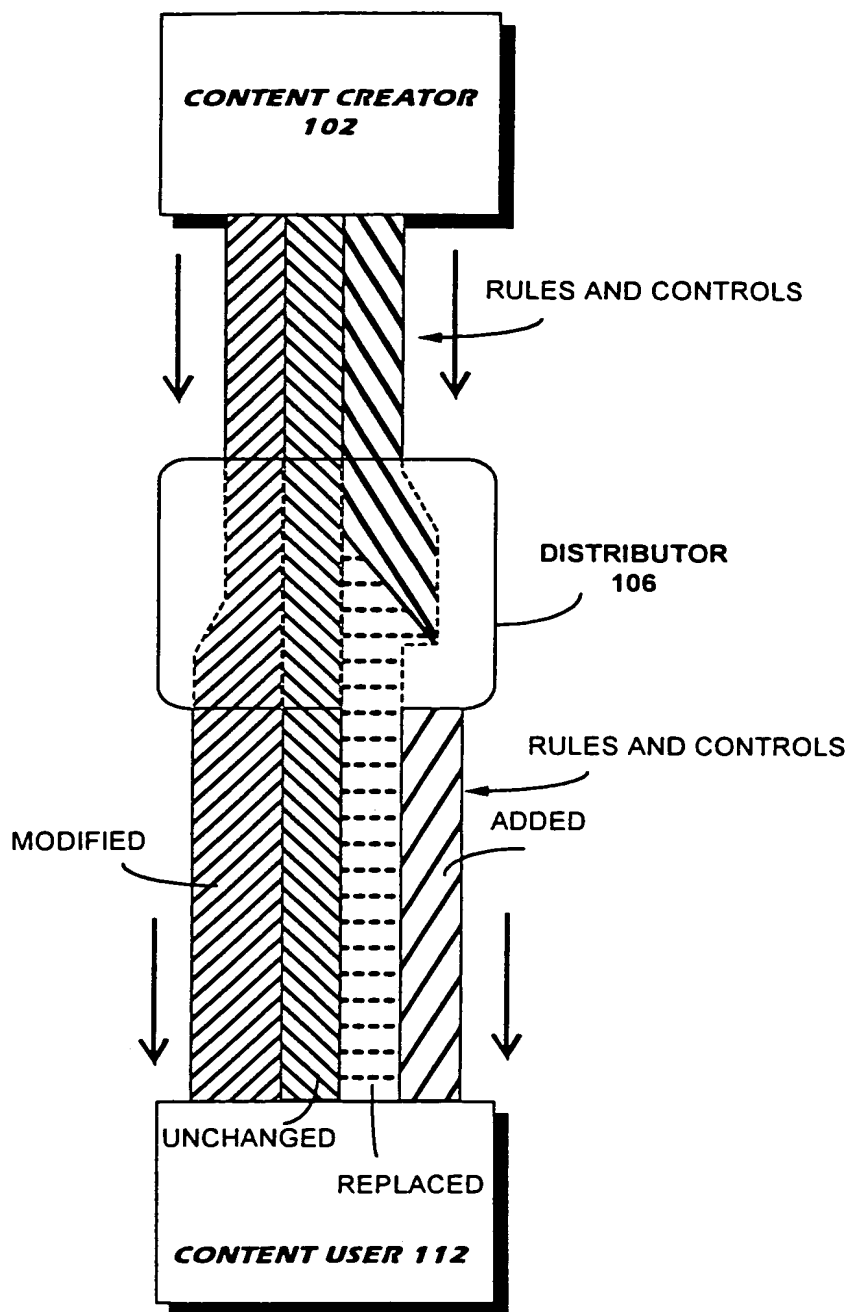
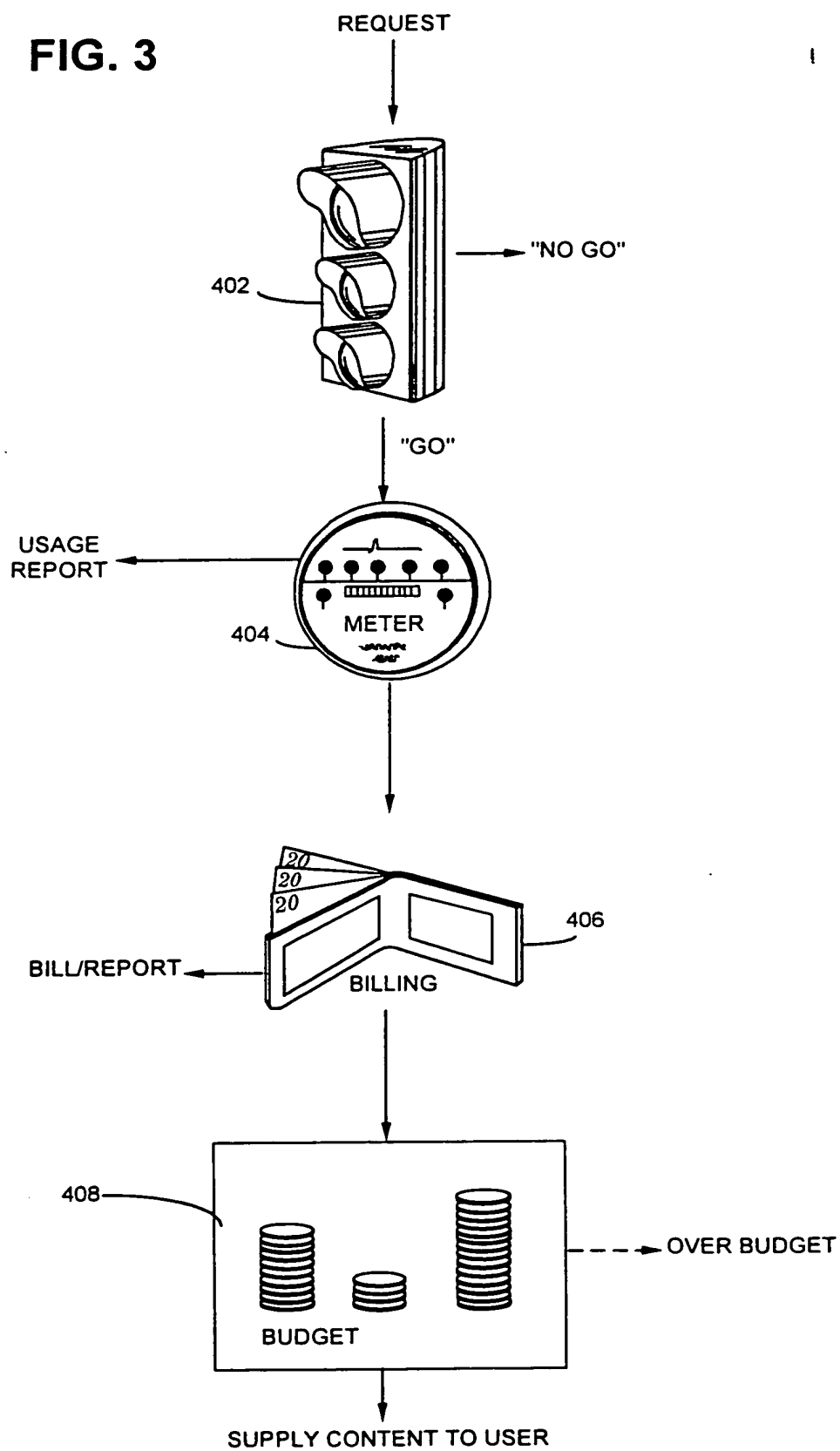


FIG. 2A



036762552 1003300

FIG. 3



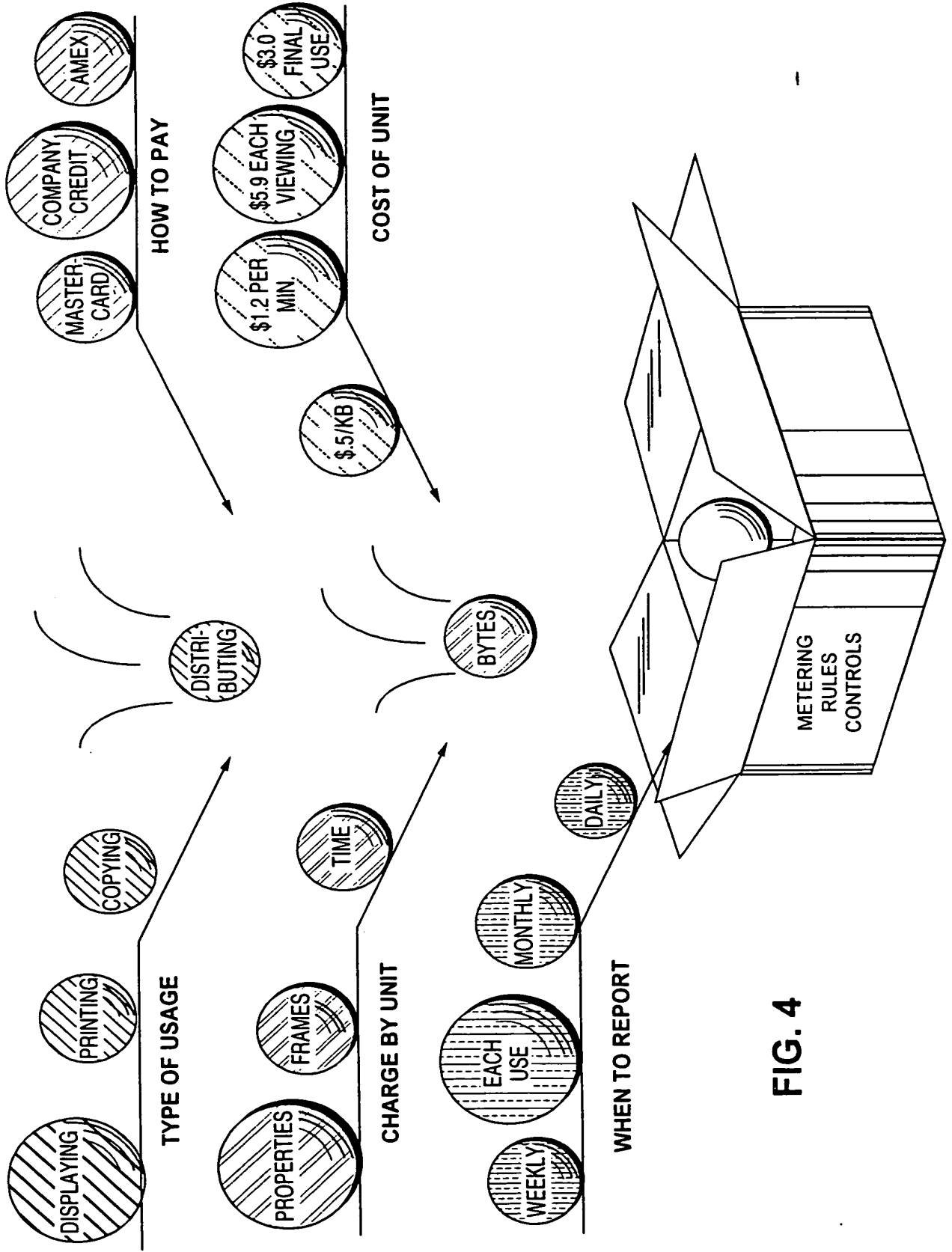
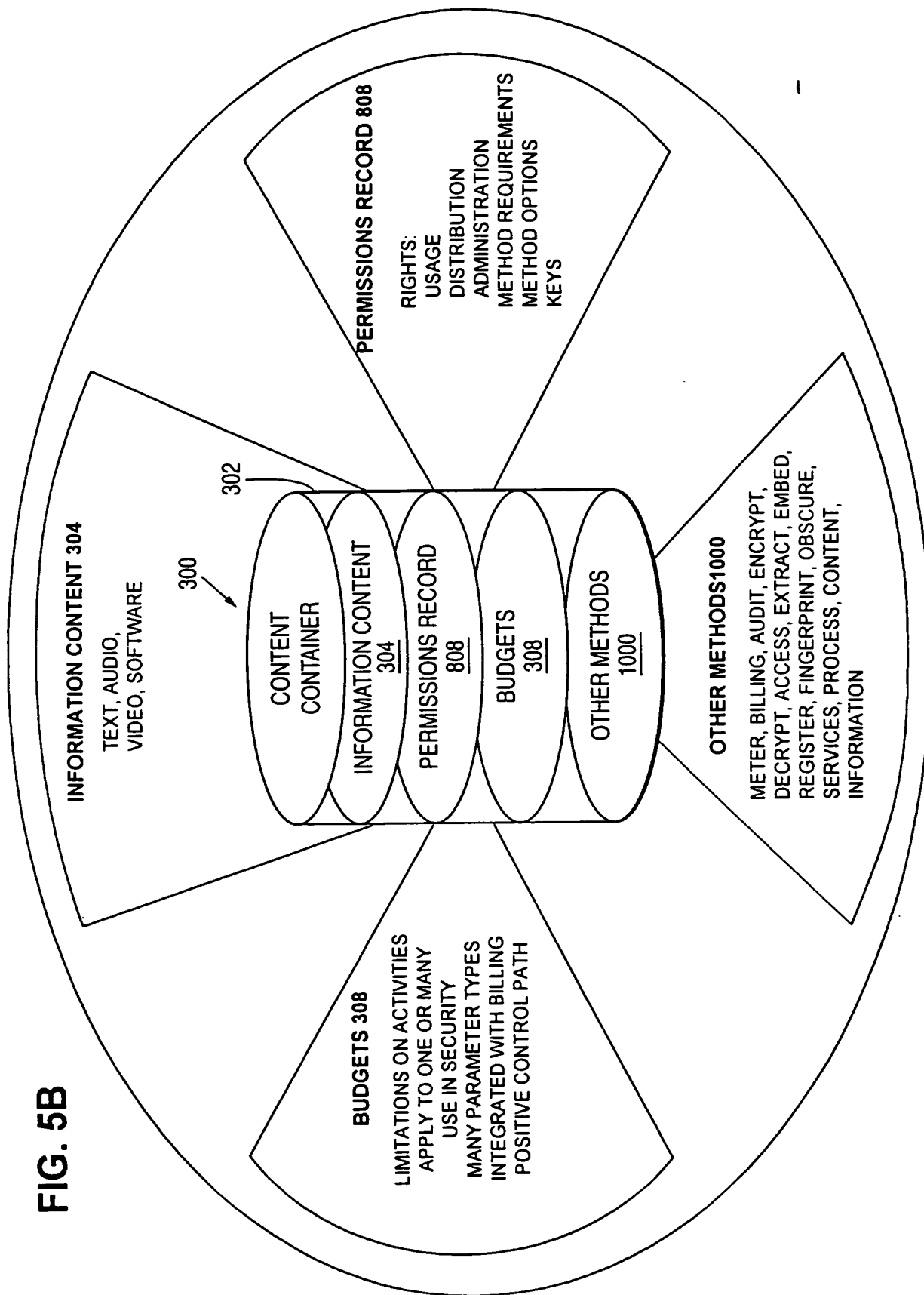


FIG. 4

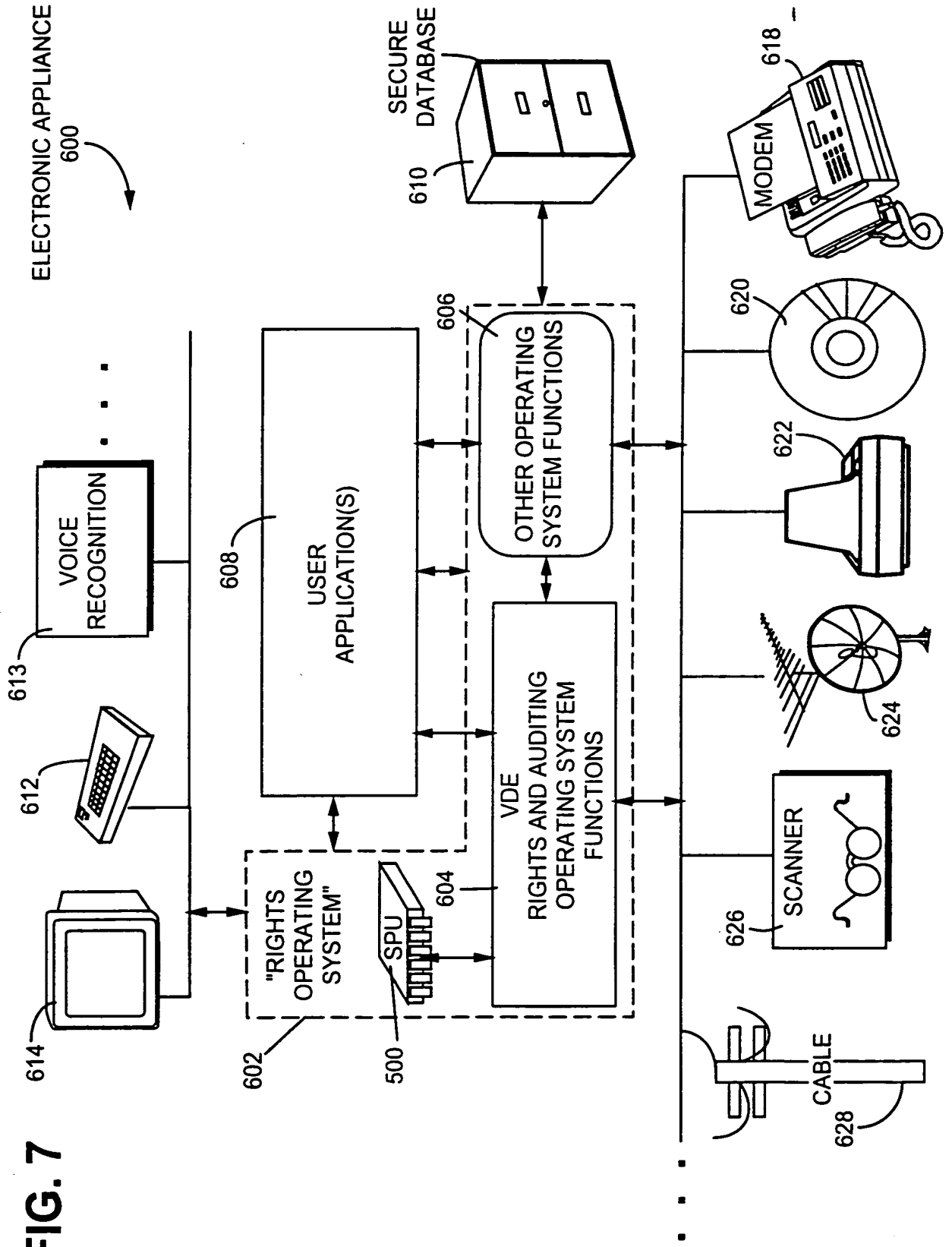
FIG. 5B





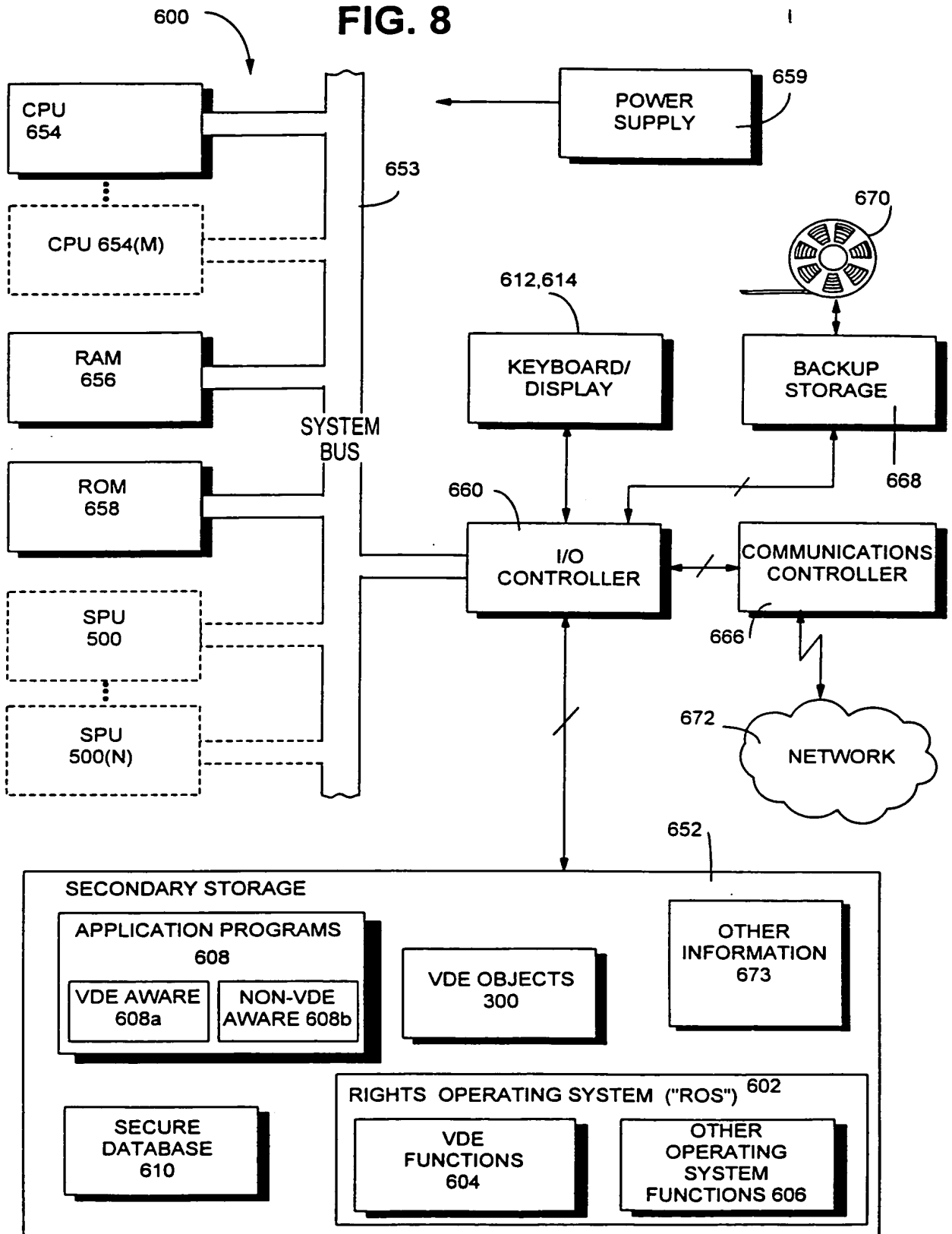
The diagram illustrates a secure processing environment (503) enclosed within a tamper-resistant barrier (502). An appliance link (510) is connected to the environment. Inside, a secure processing unit (SPU 500) contains firmware (508) and hardware (506). The barrier features a series of slots (504) on one side and a side interface (505) on the other.

FIG. 7



09676252 100300

FIG. 8



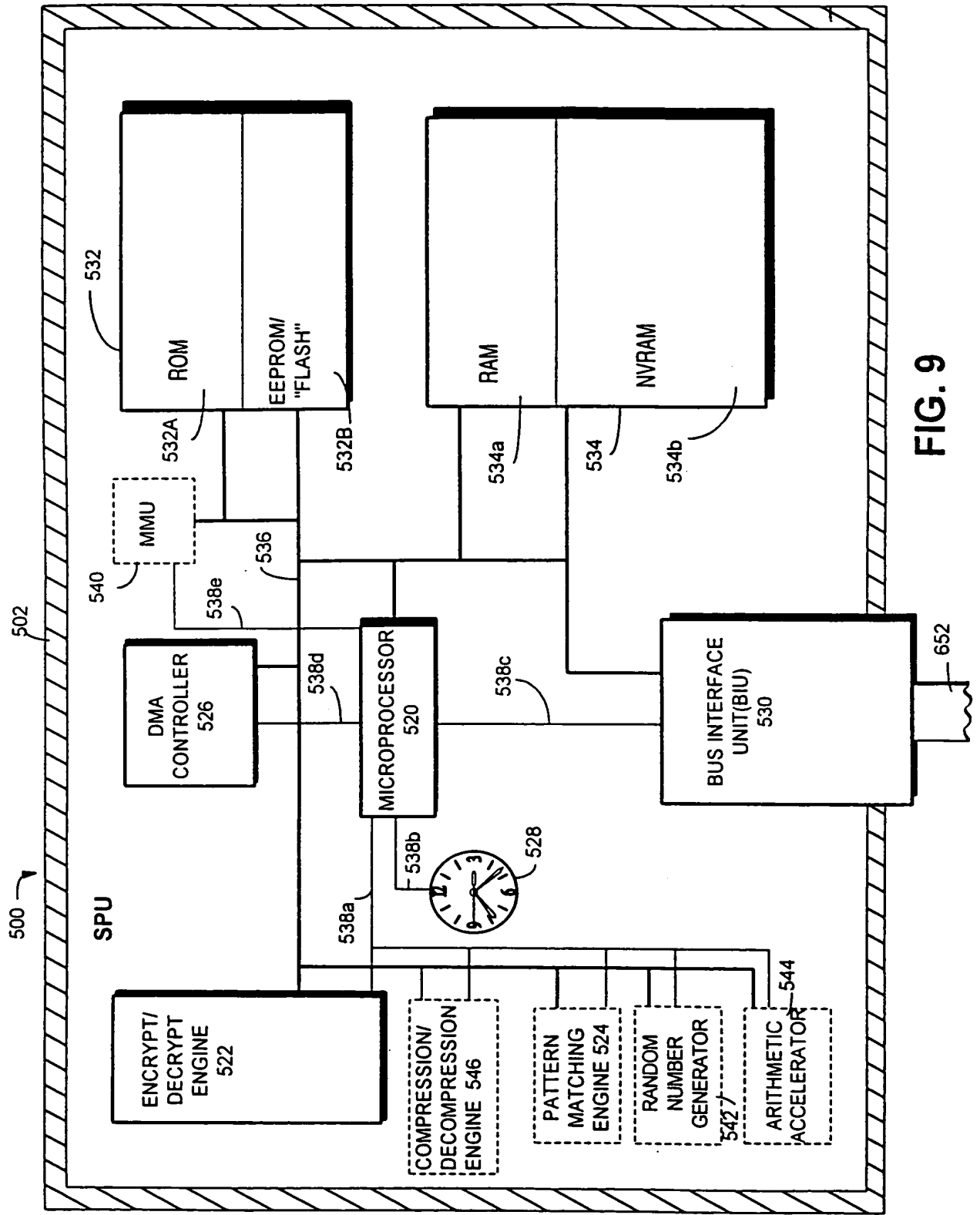


FIG. 9

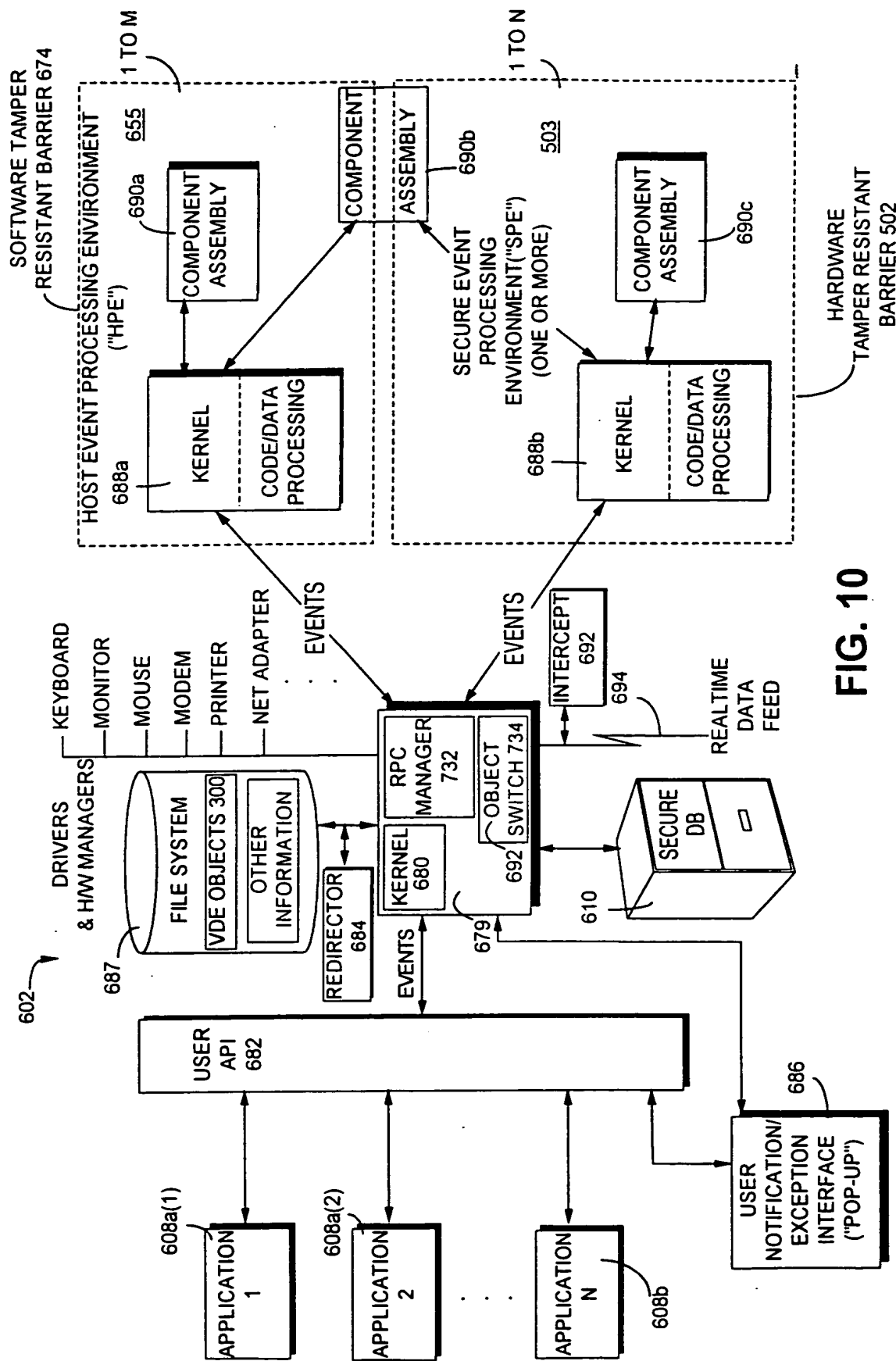


FIG. 10

FIG. 11C

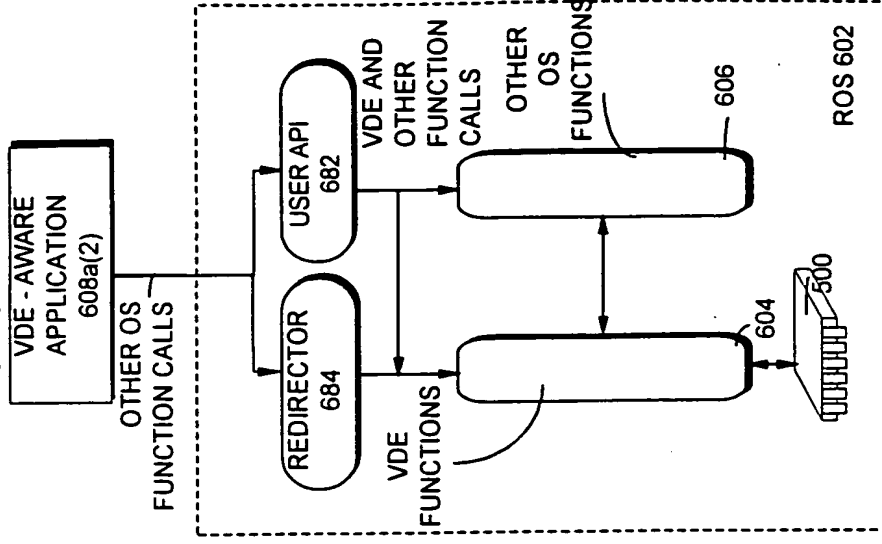


FIG. 11B

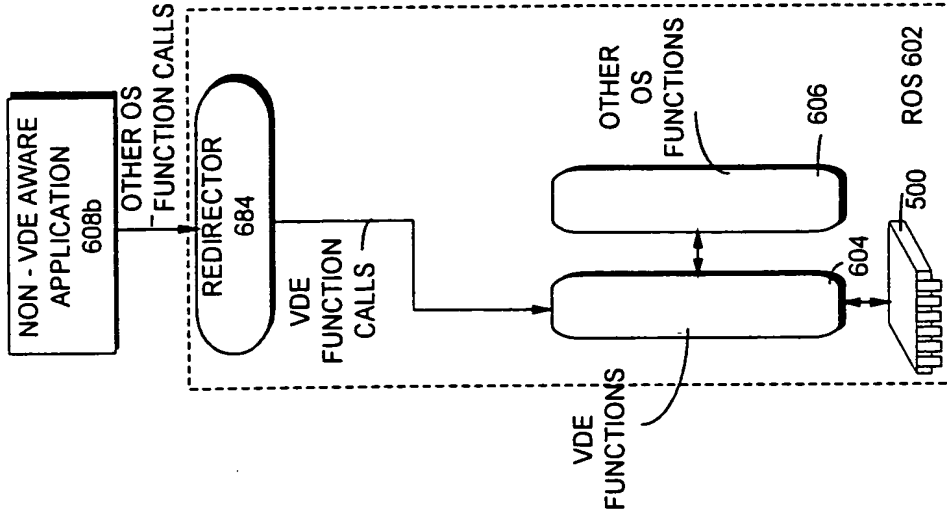
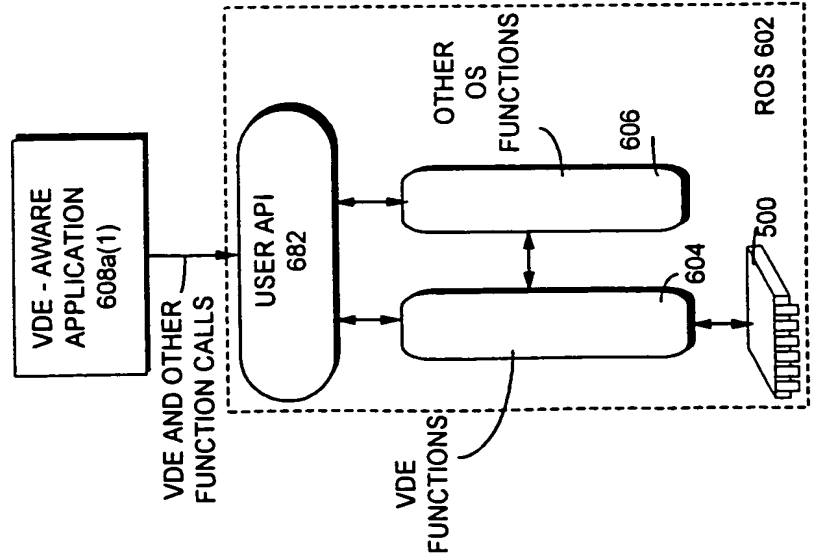


FIG. 11A



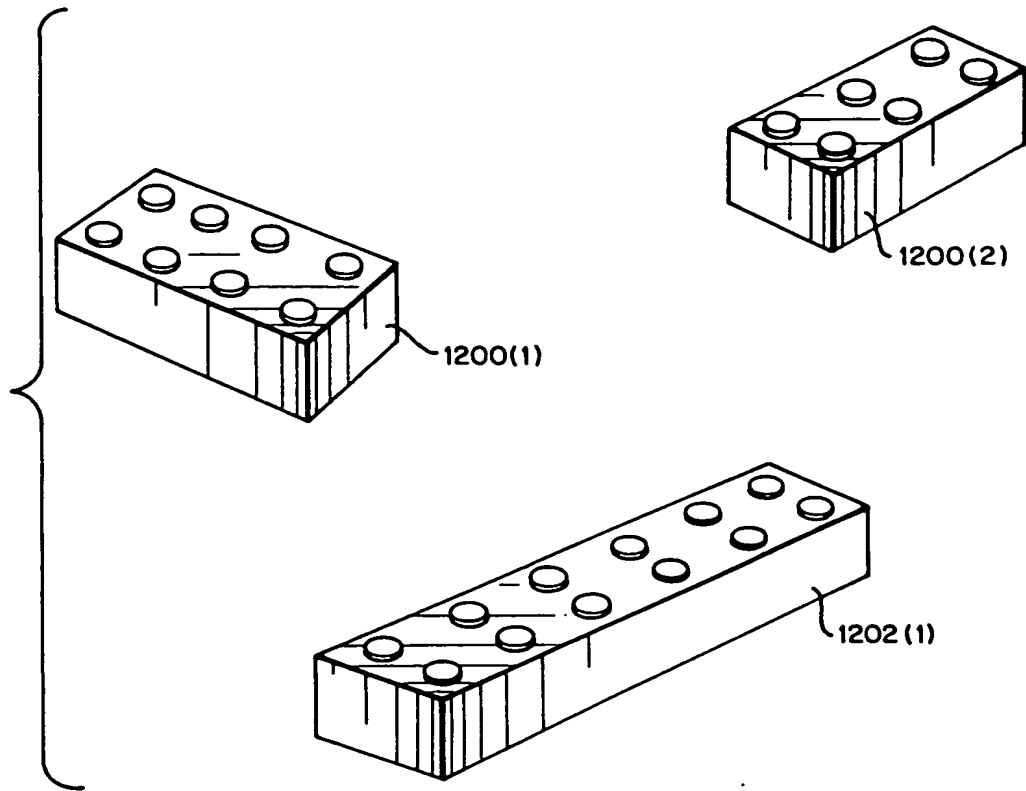


FIG. 11H

09676252 100500

FIG. 111

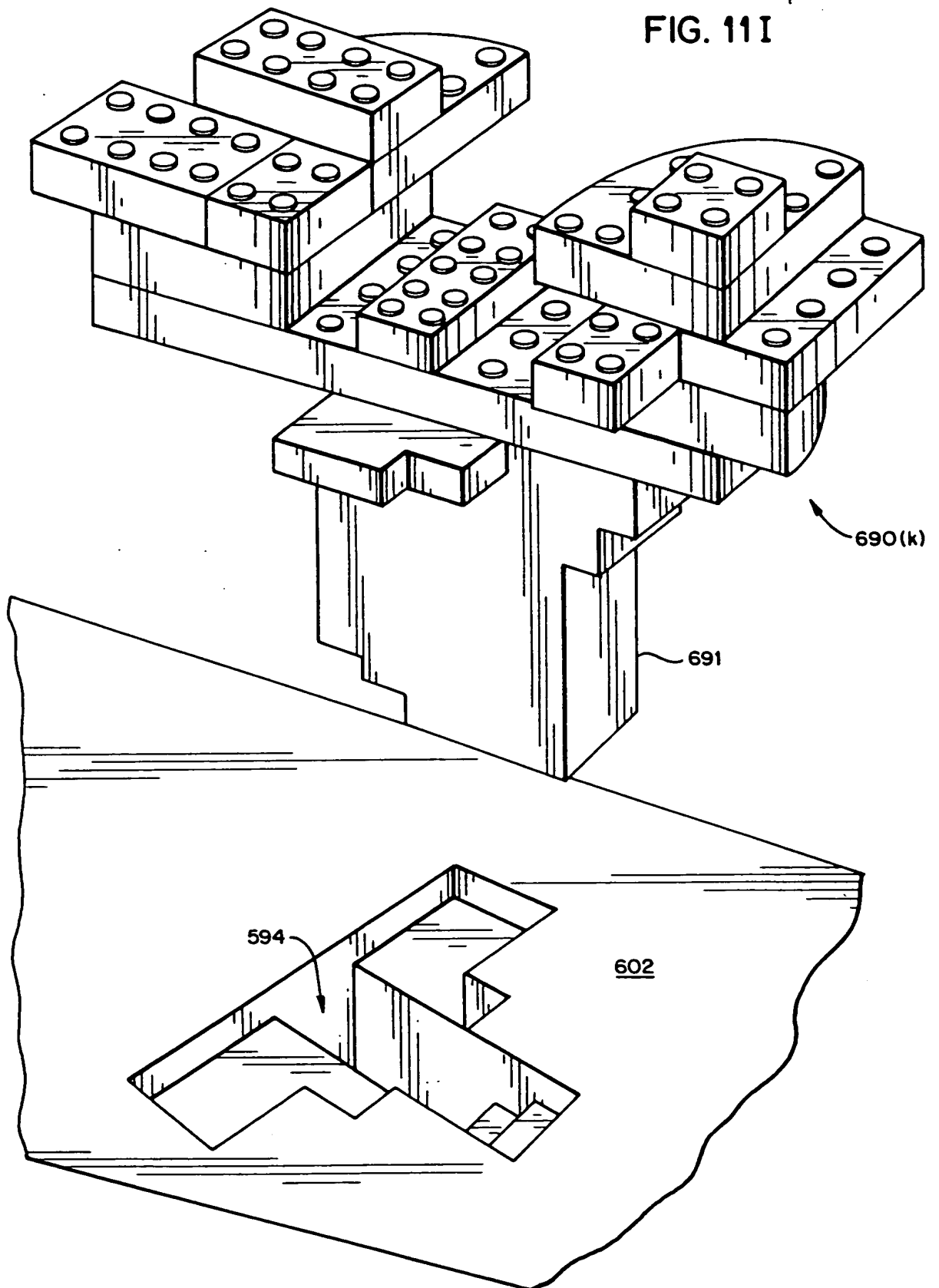




FIG. 11J

FIG. 12

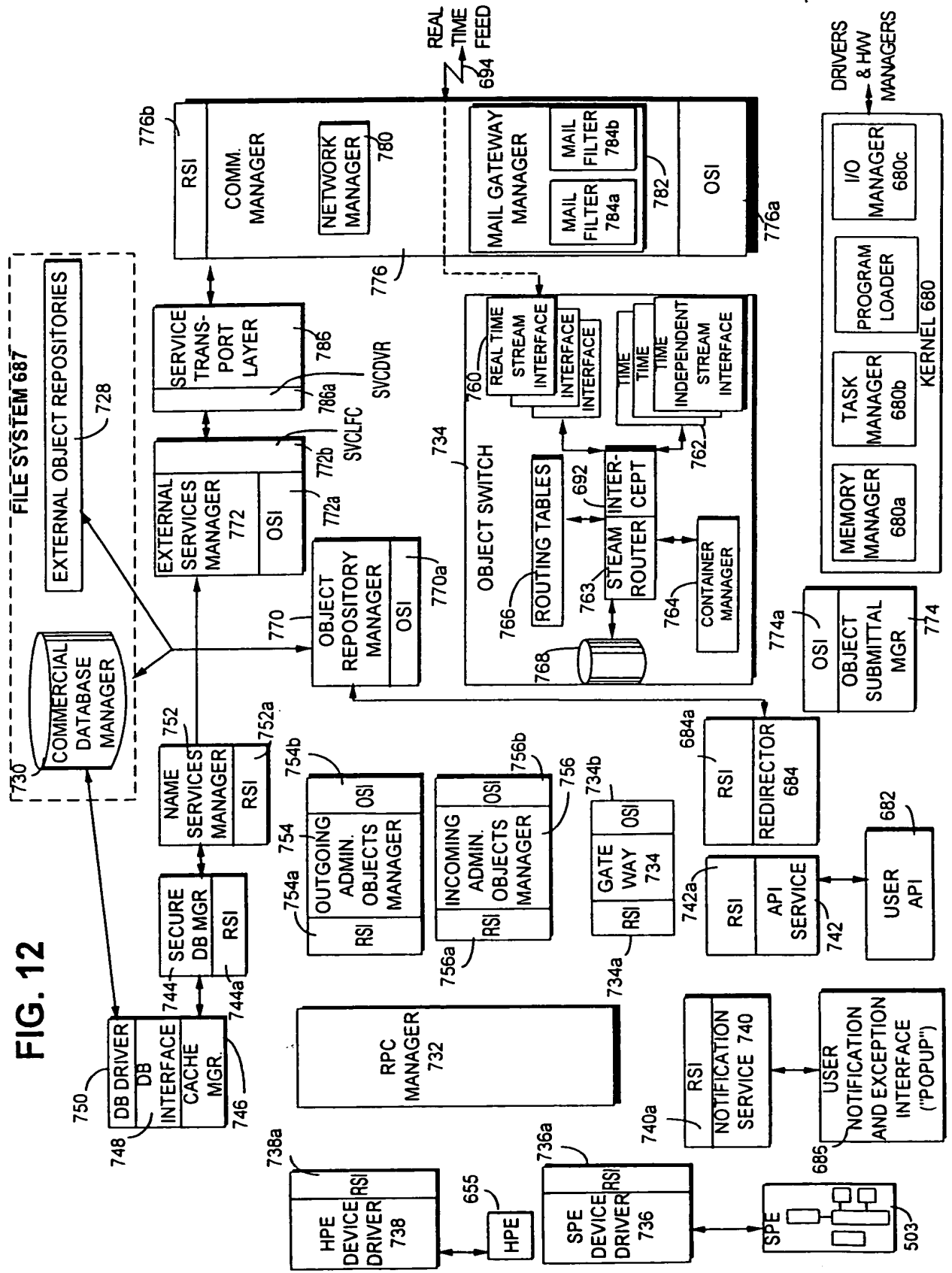


FIG. 14A

DEVICE FIRM WIRE LOW LEVEL SERVICES 582
INITIALIZATION
POST
DOWNLOAD CHALLENGE/RESPONSE AND AUTHENTICATION
RECOVERY
EEPROM/FLASH MEMORY MANAGER
KERNEL/DISPATCHER 552
INITIALIZATION
TASK MANAGER 576 (SLEEP/AWAKE/CONTEXT SWAP)
INTERRUPT HANDLER 584 (TIMER/BIU/POWER FAIL/WATCHDOG TIMER/ENCRYPTION COMPLETED)
BIU HANDLER 586
MEMORY MANAGER 578
INITIALIZATION (SETTING MMU TABLES
ALLOCATE
DEALLOCATE
VIRTUAL MEMORY MANAGER 580
SWAP BLOCK PAGING
EXTERNAL MODULE PAGING
MEMORY COMPRESS
RPC AND TABLES 550
INITIALIZATION
MESSAGING CODE /SERVICES MANAGER
SEND/RECEIVE
STATUS
RPC DISPATCH TABLE
RPC SERVICE TABLE

•
•
•

TIME BASE MANAGER 554
ENCRYPTION/DECRYPTION MANAGER 556
PK
BULK
KEY AND TAG MANAGER 558
KEY STORAGE IN EEPROM
KEY LOCATOR
KEY GENERATOR
CONVOLUTION ALGORITHM
SUMMARY SERVICES MANAGER 560
EVENT SUMMARIES
BUDGET SUMMARIES
DISTRIBUTER SUMMARY SERVICES
CHANNEL SERVICES MANAGER 562
CHANNEL HEADERS
CHANNEL DETAILS
LOAD MODULE EXECUTION SERVICES 568
AUTHENTICATION MANAGER/SECURE COMMUNICATION MANAGER 564
DATABASE MANAGER 566
MANAGEMENT FILE SUPPORT
TRANSACTION AND SEQUENCE NUMBER SUPPORT
SRN/ HASH
DTD INTERPRETER 590
LIBRARY ROUTINES 574
I/O CALLS (STRING SEARCH ETC.)
MISC. ITEMS THAT ARE PROBABLY LIBRARY ROUTINES
TAG CHECKING, MD5, CRC'S
INTERNAL LM'S 572 FOR BASIC METHODS
METER LOAD MODULE(S)
BILLING LOAD MODULE(S)
BUDGET LOAD MODULE(S)
AUDIT LOAD MODULE(S)
READ OBJECT LOAD MODULE(S)
WRITE OBJECT LOAD MODULE(S)
OPEN OBJECT LOAD MODULE(S)
CLOSE OBJECT LOAD MODULE(S)

•
•
•

09670252 " 100300

FIG. 14B

•
•
•

PUBLIC KEY AND PRIVATE KEY, SYSTEM ID, AUTHENTICATION CERTIFICATE,VDE SYSTEM PUBLIC KEY, PRIVATE DES KEY
TOP LEVEL KEYS FOR OBJECTS
TOP LEVEL BUDGET INFO
METER SUMMATION VALUES
KEY RECORDS FOR BUDGET RECORDS, AUDIT RECORDS, STATIC MANAGEMENT RECORDS, UPDATED MANAGEMENT RECORDS, ETC.
• • •
DEVICE DATA TABLE
SITE ID
TIME
ALARMS
TRANSACTION/SEQUENCE #'S
MISCELLANEOUS
MEMORY MAP
MAP METERS
LM/UDT TABLE
TASK MANAGER 576
CHANNEL(S)
SUMMARY SERVICES 560
SECURE DATABASE TAGS
SRN ENTRIES
HASH ENTRIES

•
•
•

09670252 = 400300

03670252 100500

FIG. 14C

STACK	
•	
CHANNEL SWAP BLOCK	CHANNEL LM
	CHANNEL HEADER & D1
CONTROL SWAP BLOCK	CONTROL LM
	CONTROL D1
	COMMIT LM
	COMMIT D1, D2, D3
EVENT SWAP BLOCK	EVENT LM
	MAP TABLE (SINGLE) D1
METER SWAP BLOCK	METER LM
	METER UDE DELTA,DELTA'
	METER TRAIL LM
	METER TRAIL UDE DELTA,DELTA'
BUDGET SWAP BLOCK	METER LM
	METER UDE DELTA,DELTA'
	METER TRAIL LM
	METER TRAIL UDE DELTA,DELTA'
BILLING SWAP BLOCK	BILLING LM
	METER UDE
	BUDGET UDE
	BILLING TABLE UDE
	BILLING TRAIL LM
	BILLING TRAIL UDE DELTA'

•

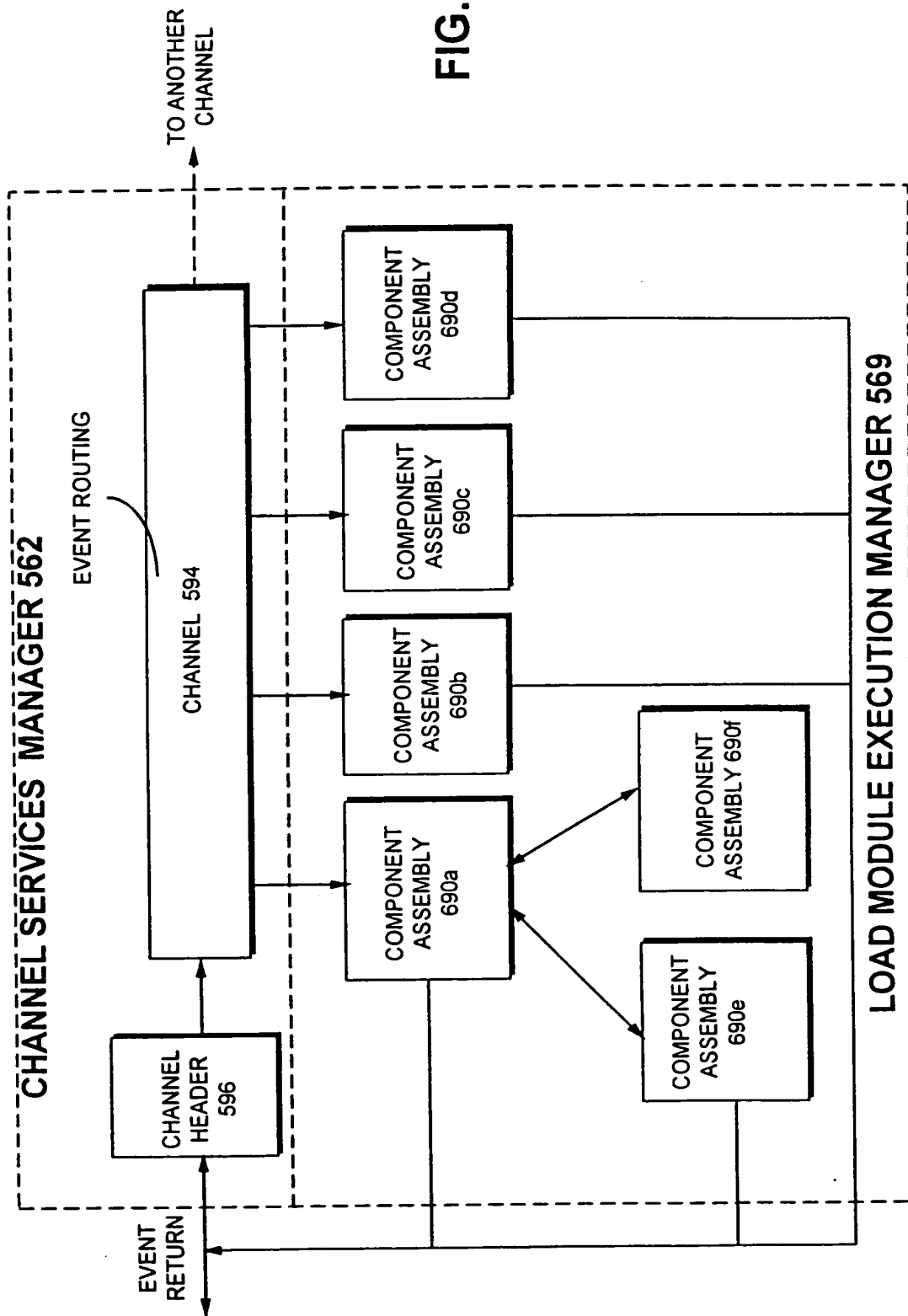


FIG. 15

FIG. 15A

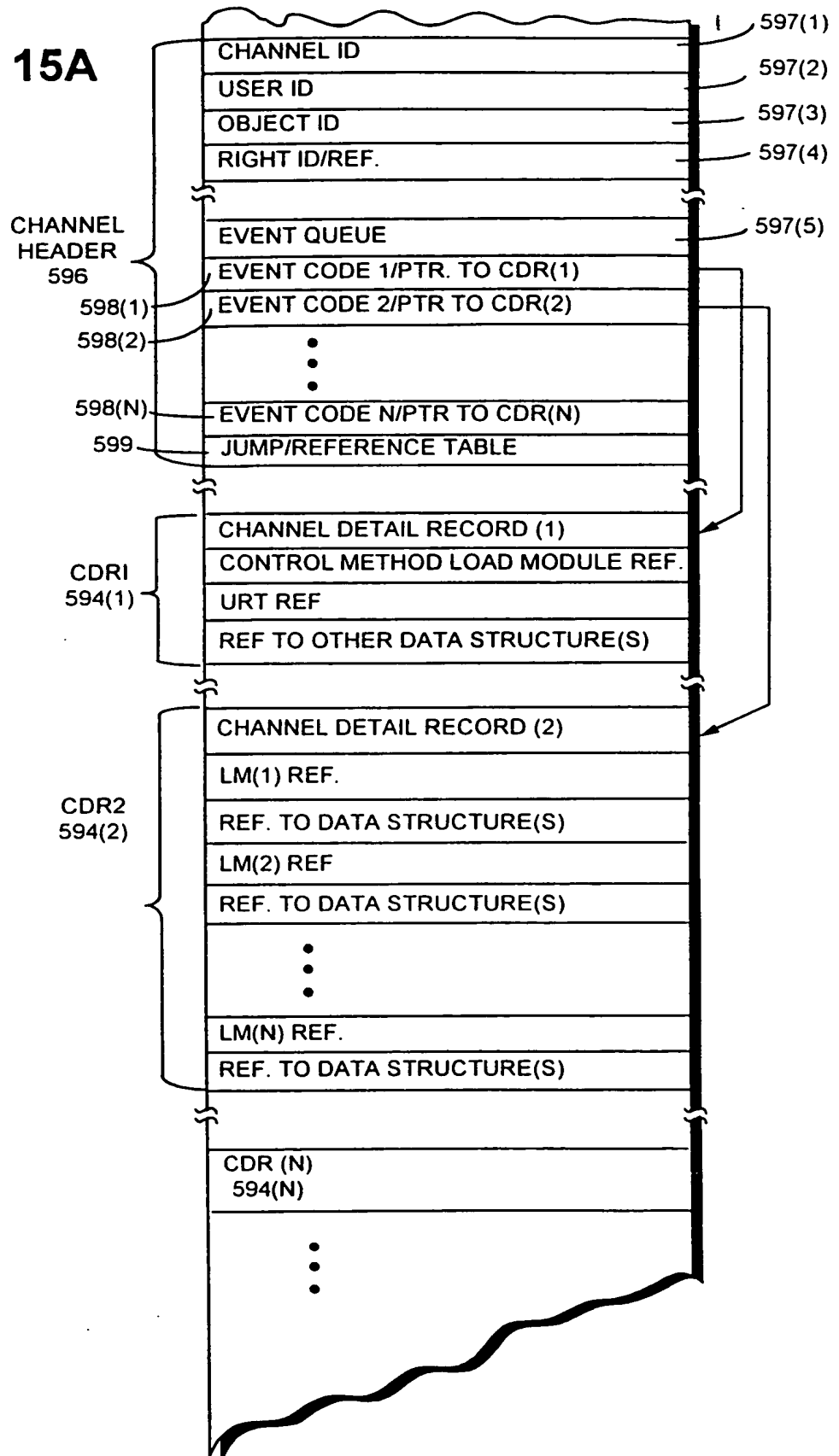
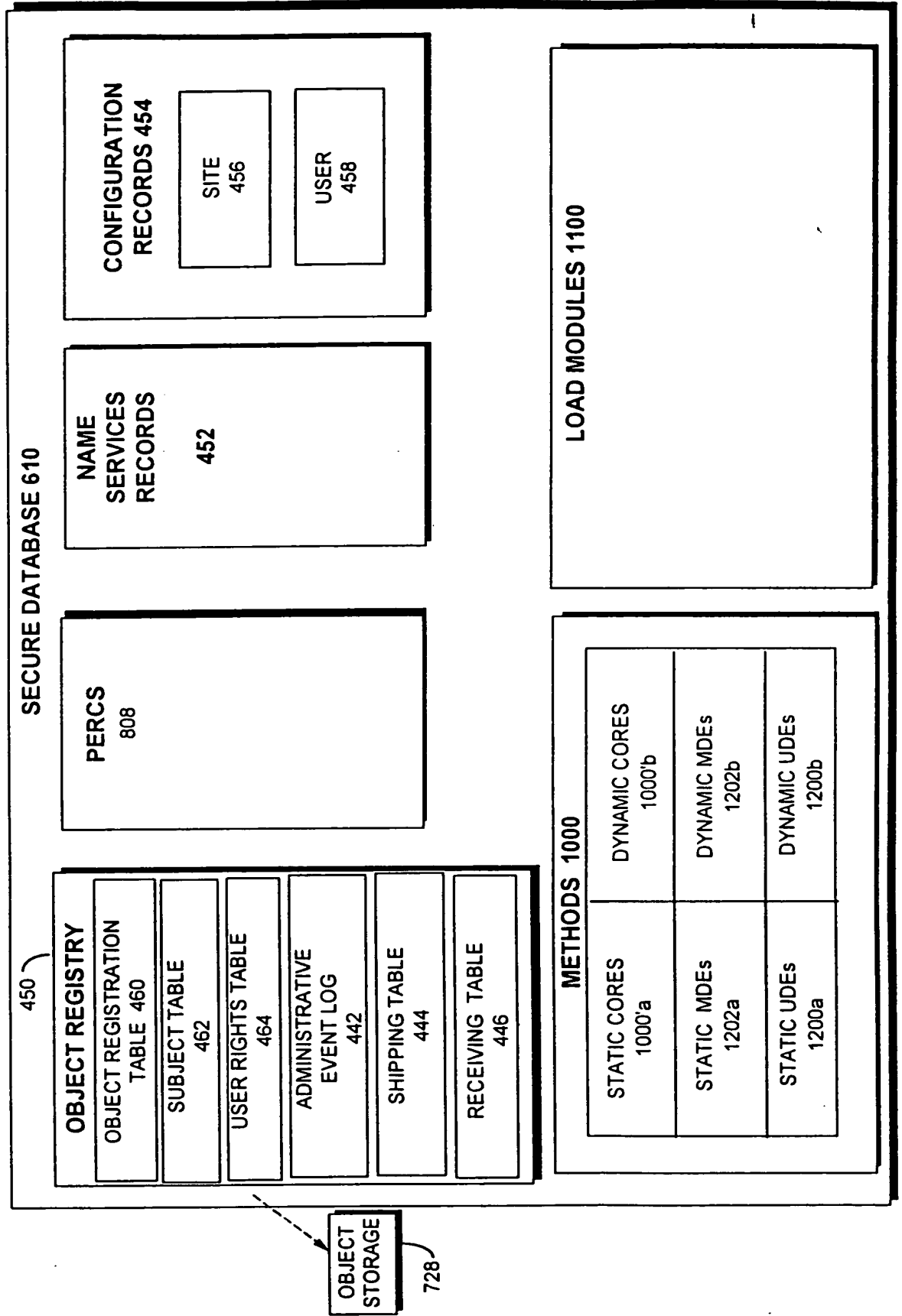


FIG. 16



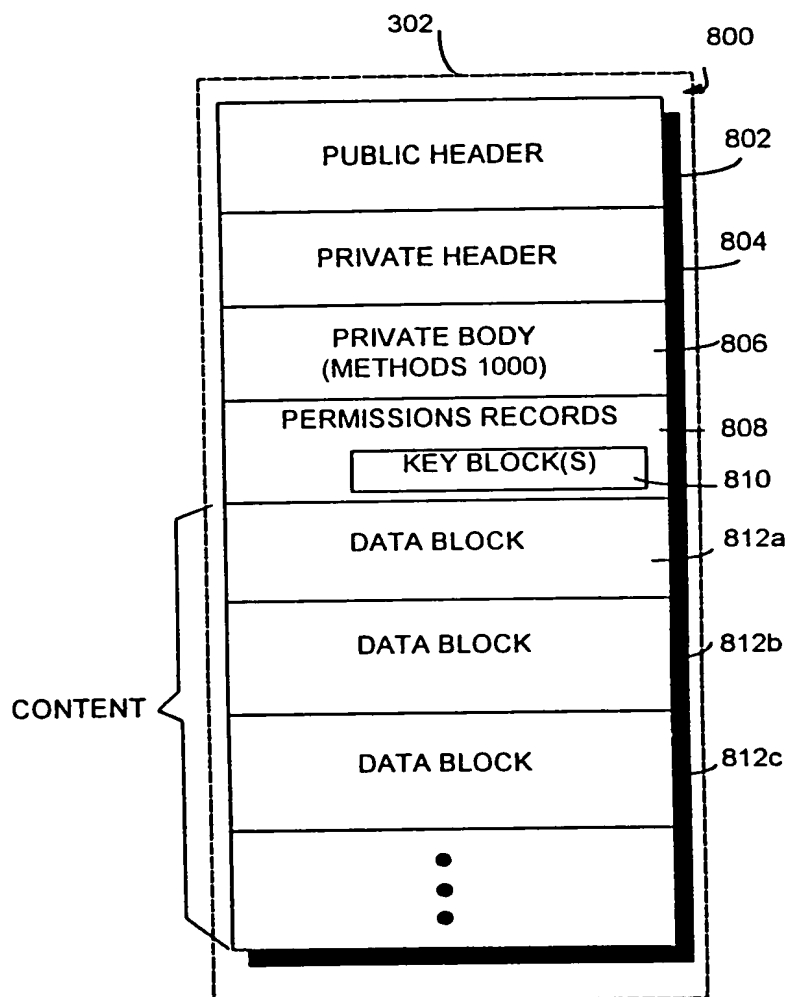


FIG. 17

00070250 100300

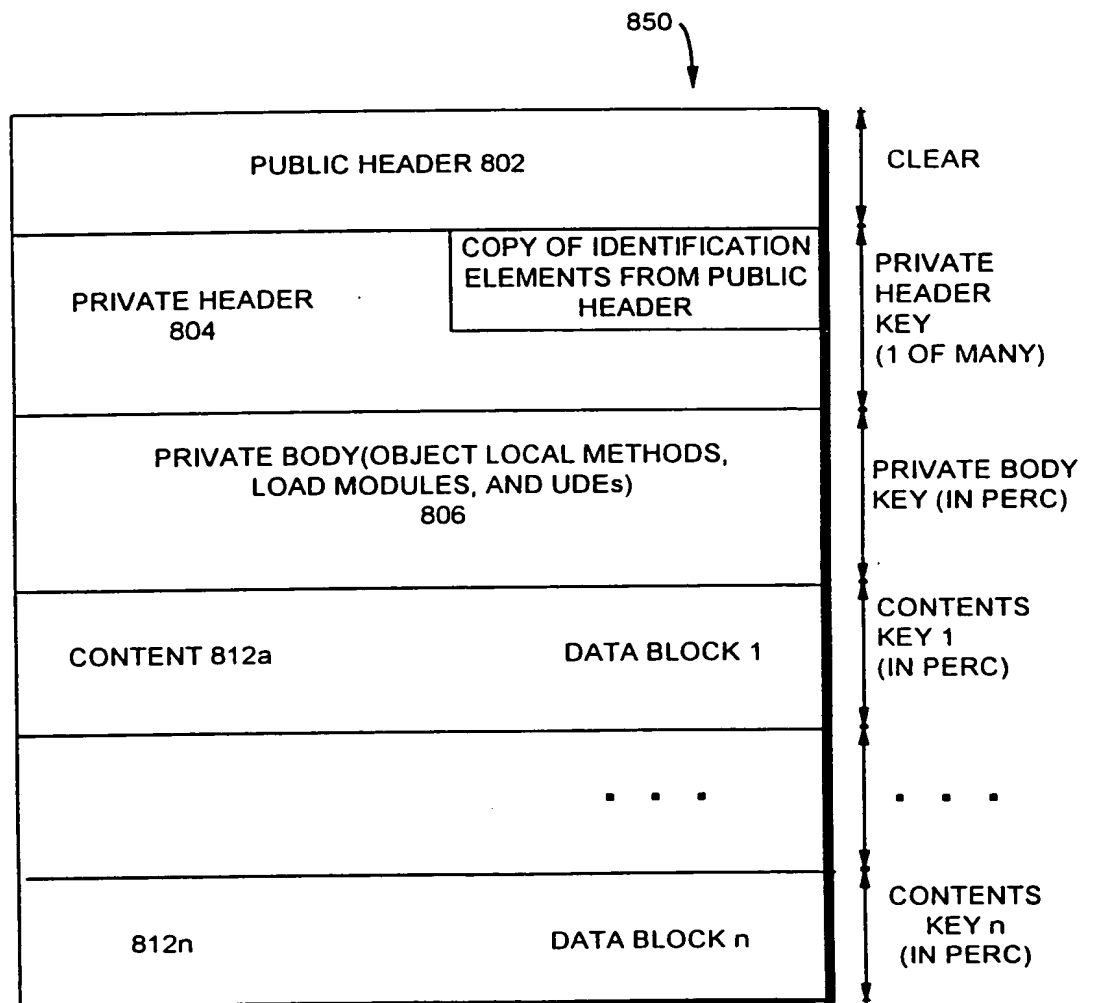


FIG. 18

860

PUBLIC HEADER 802

PRIVATE HEADER 804

COPY OF IDENTIFICATION ELEMENTS FROM PUBLIC HEADER

808

PERC

KEY BLOCKS 810

PRIVATE BODY(OBJECT METHODS, LOAD MODULES, AND UDEs) 806

CONTENT 812a

DATA BLOCK 1

...

812n

DATA BLOCK n

CLEAR

PRIVATE HEADER KEY (1 OF MANY)

PRIVATE BODY KEY (IN PERC)

CONTENTS KEY 1 (IN PERC)

...

CONTENTS KEY n (IN PERC)

1

03070552 400300

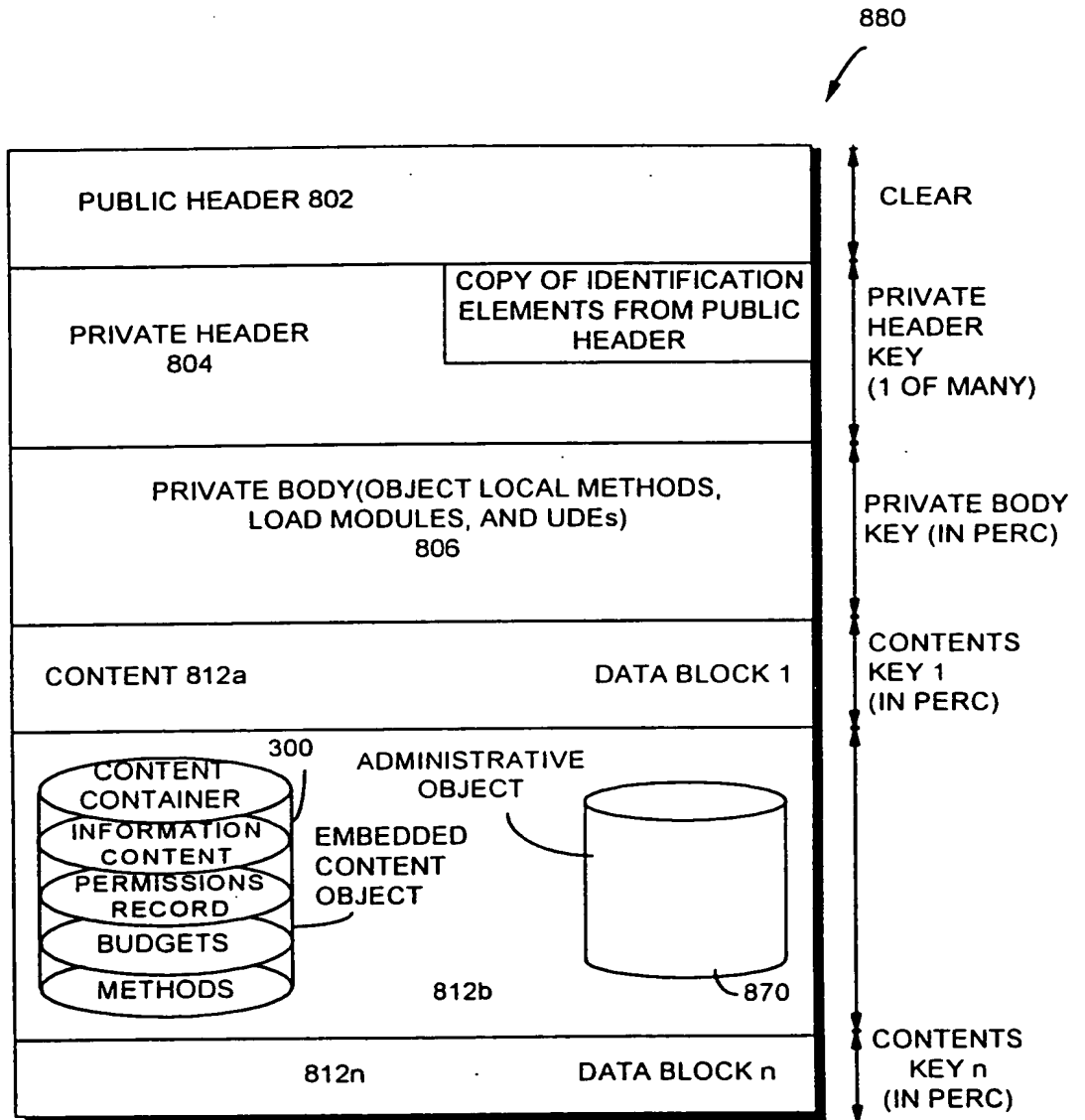


FIG. 20

FIG. 23

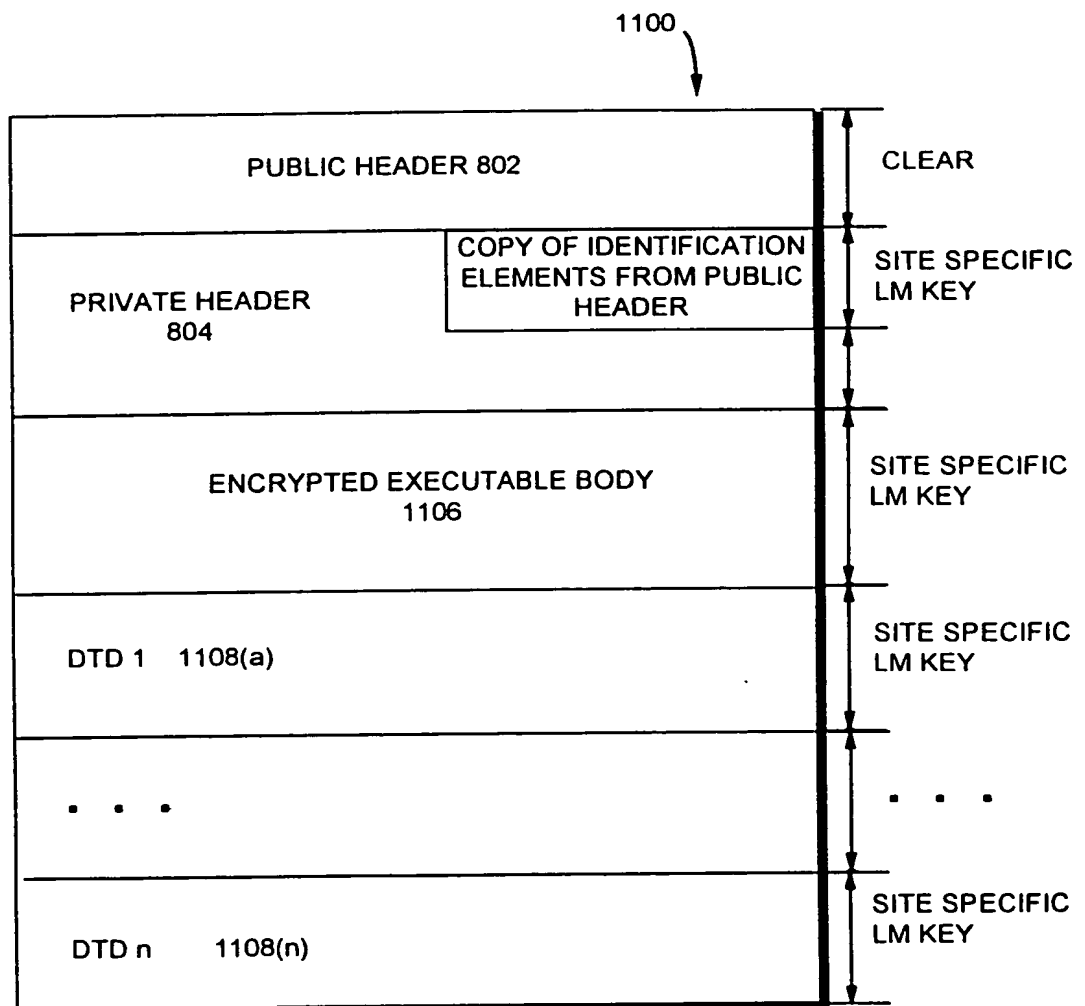
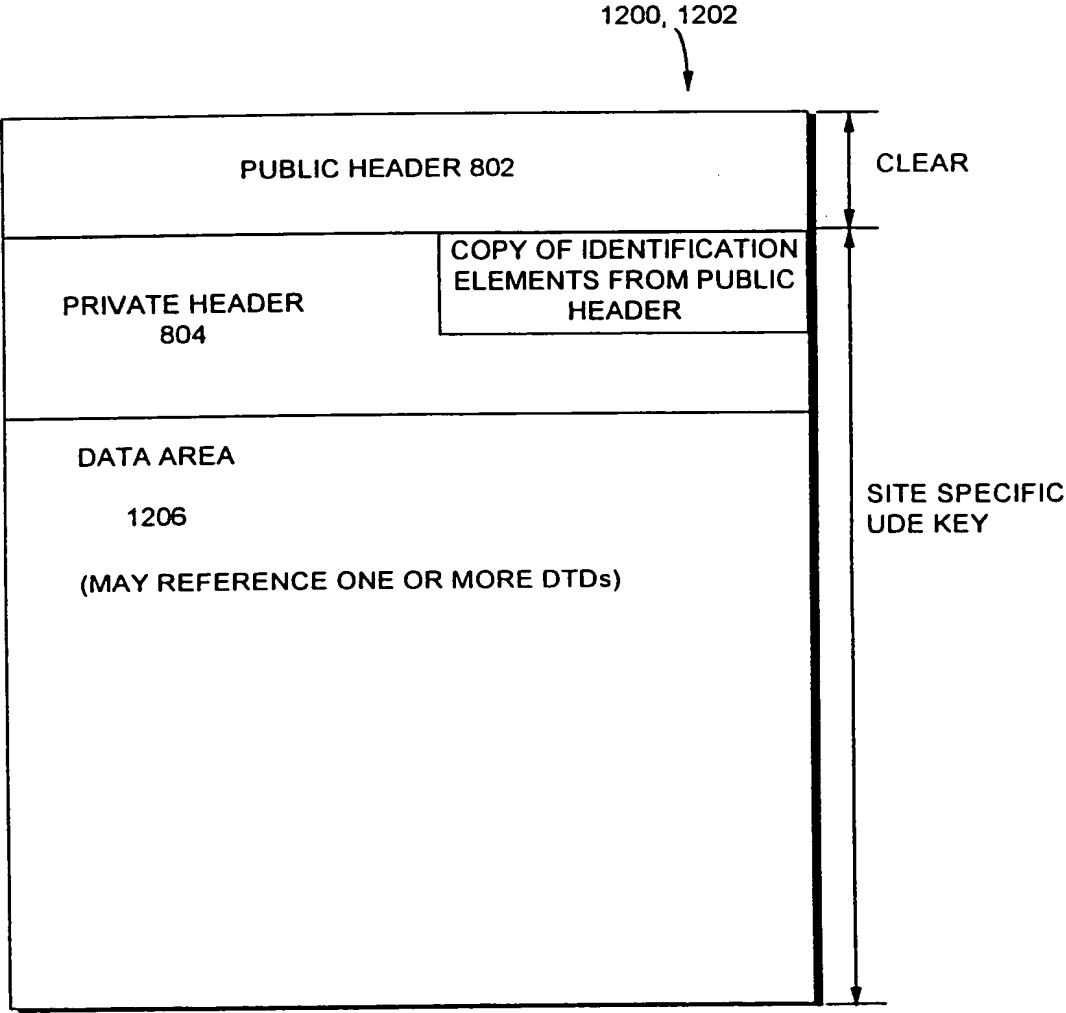


FIG. 24



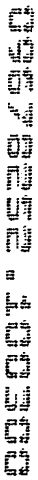
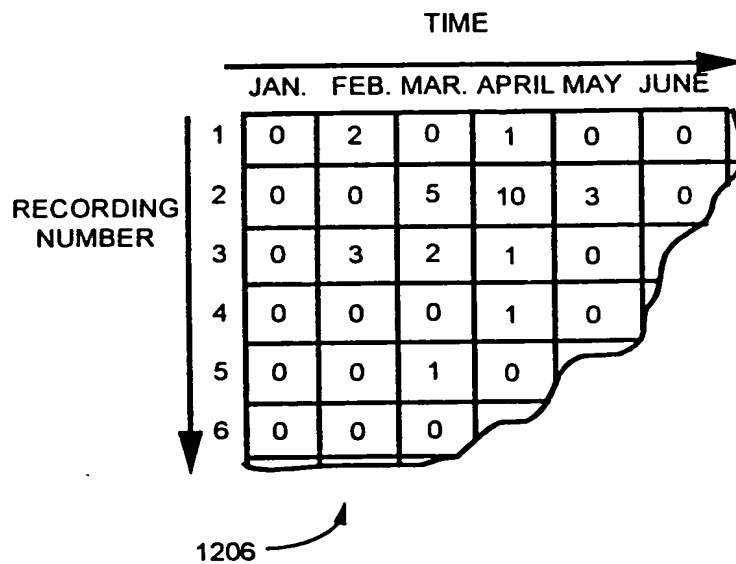
[illegible]

FIG. 25B



09676252 100300

FIG. 25C

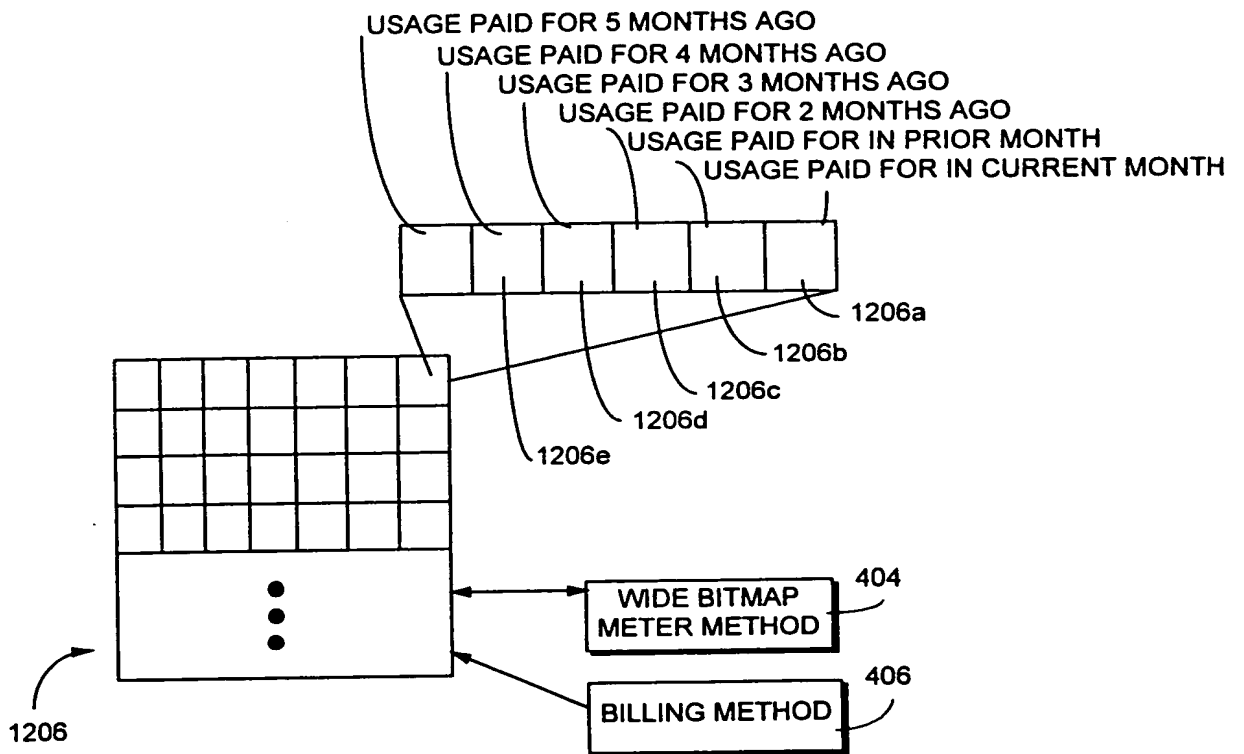


FIG. 26

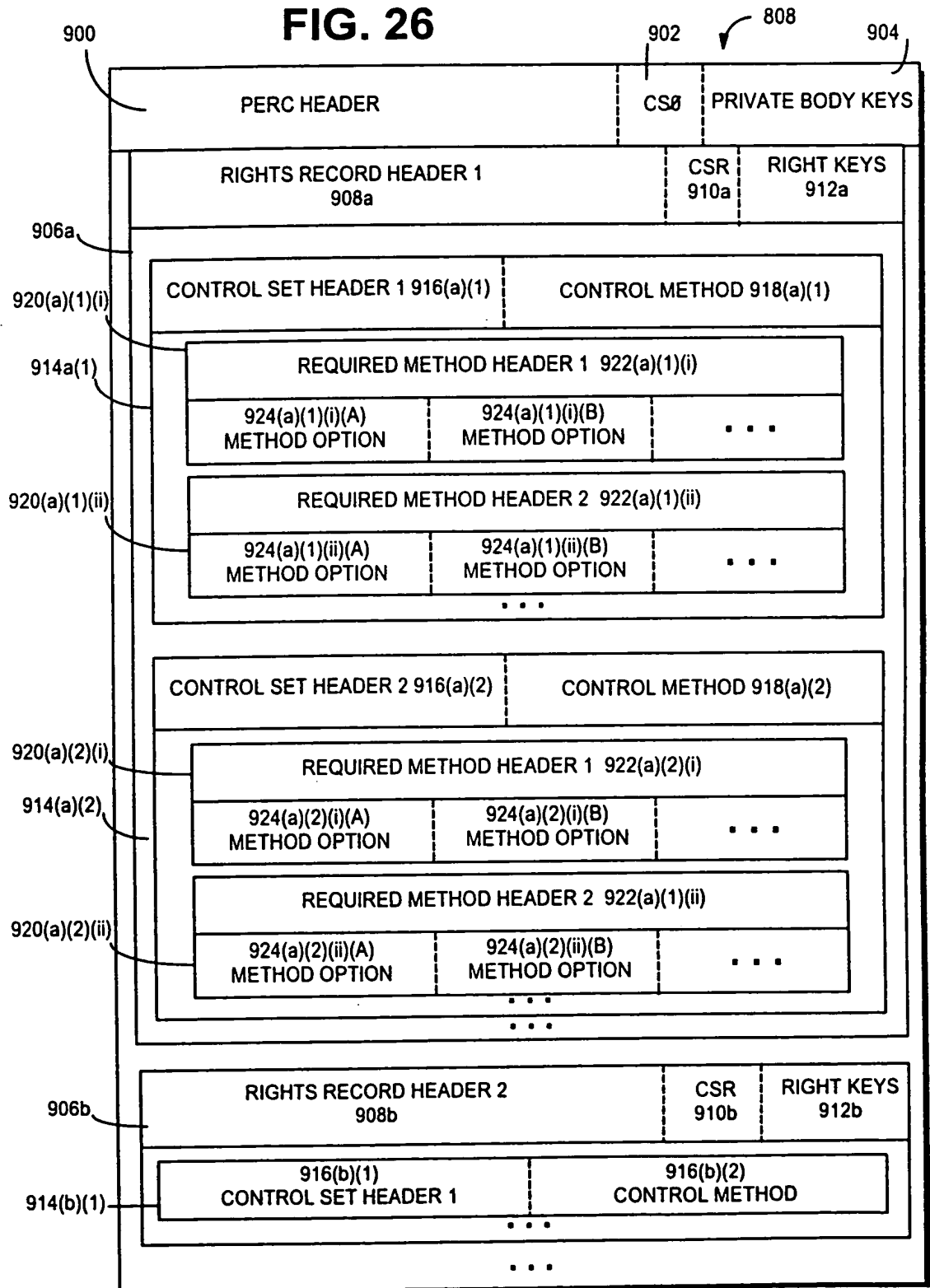
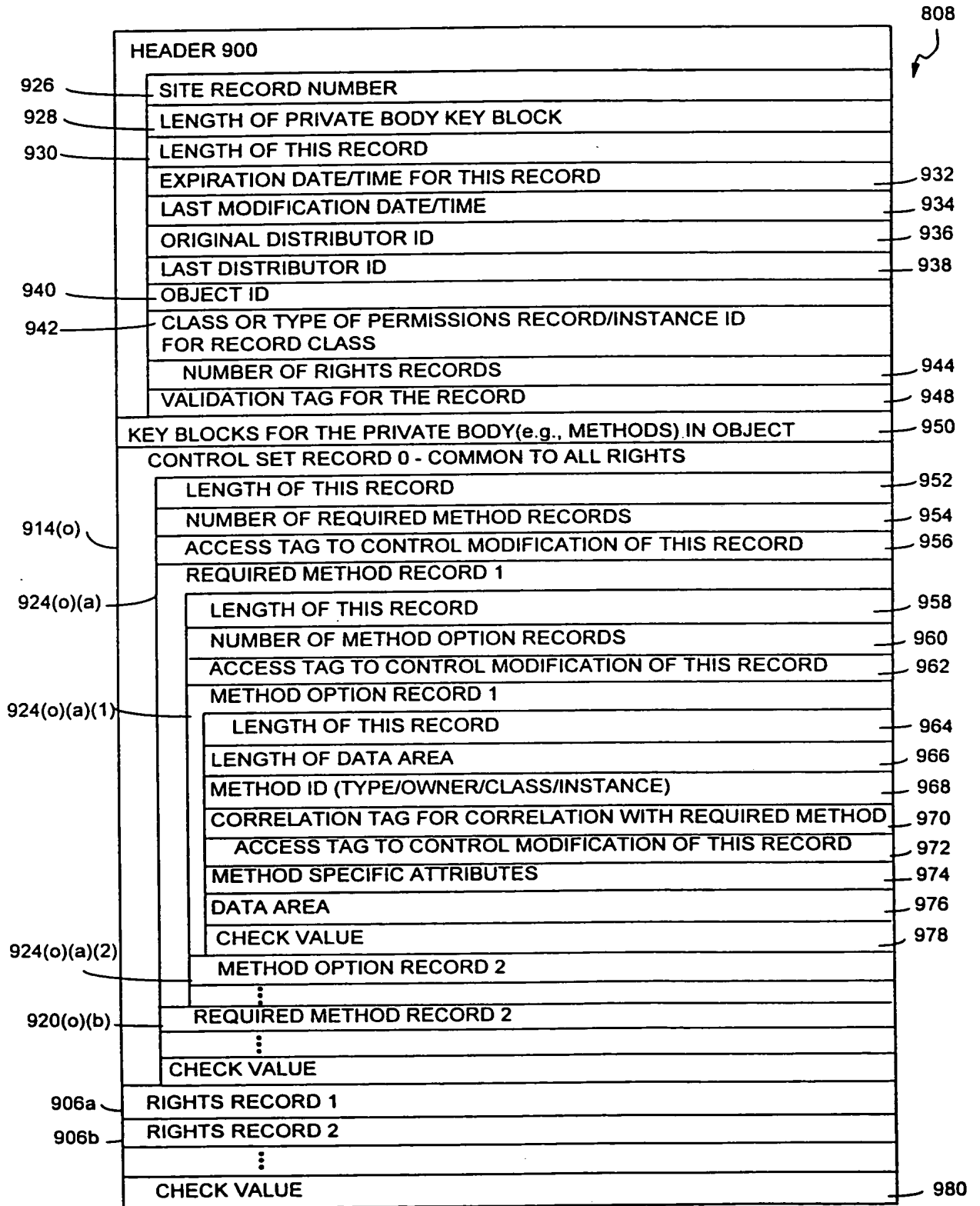


FIG. 26A



09070252 100300

FIG. 26B

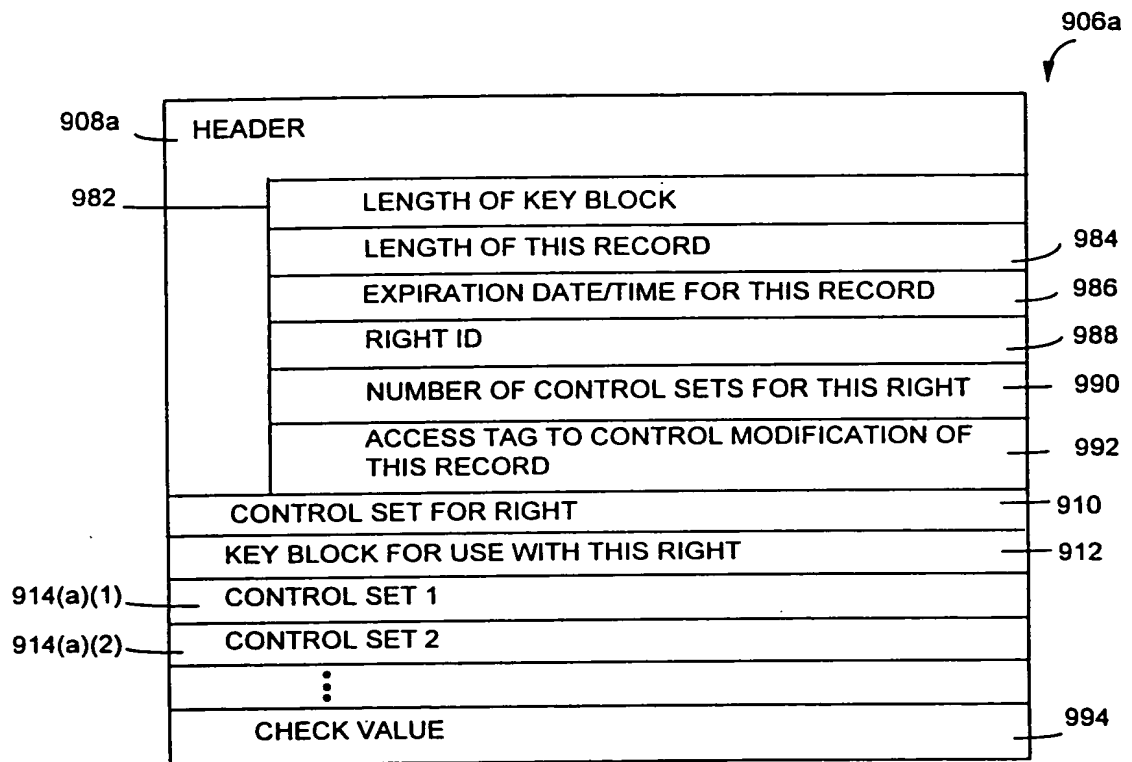
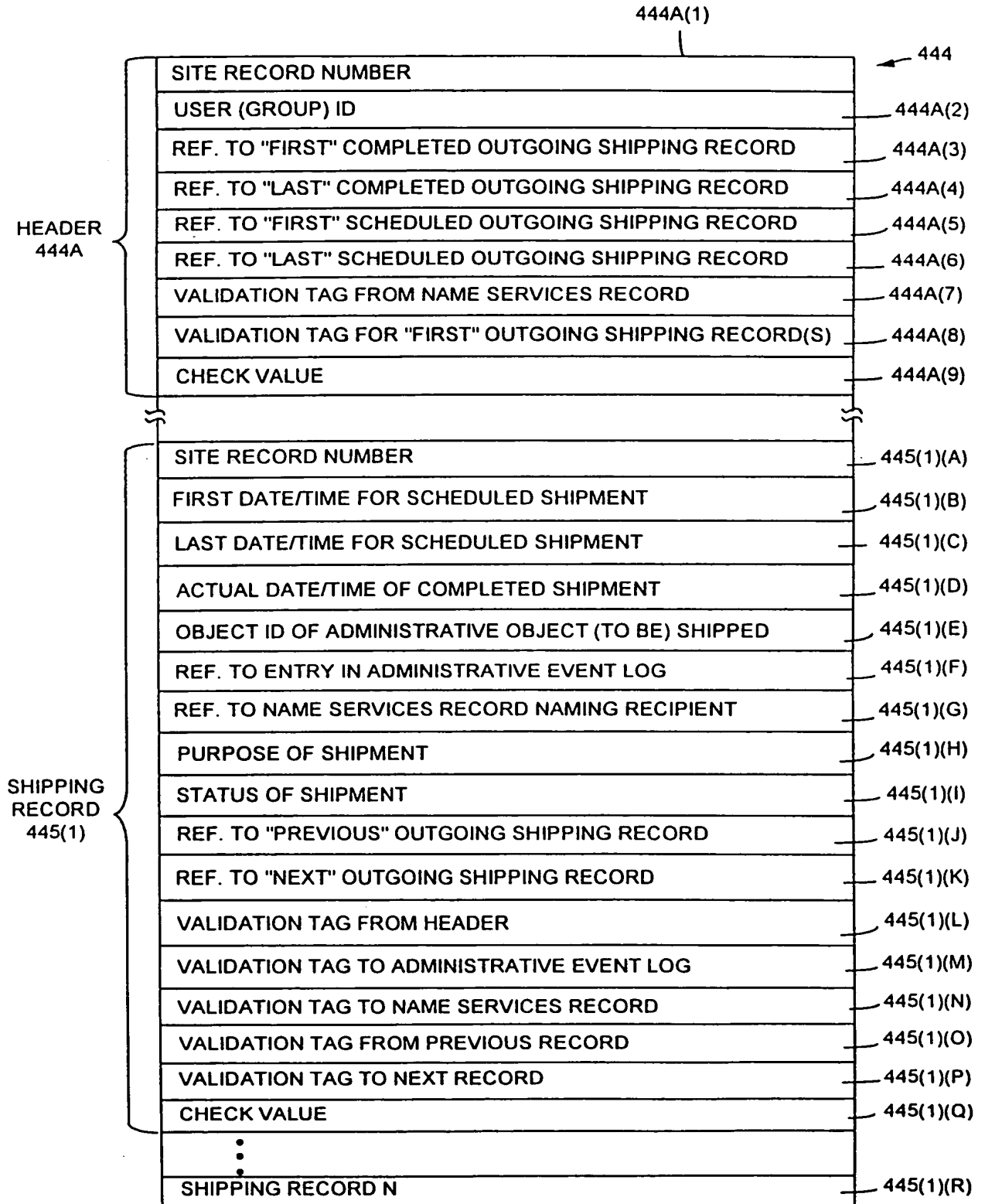


FIG. 27



05678999 103300

FIG. 28

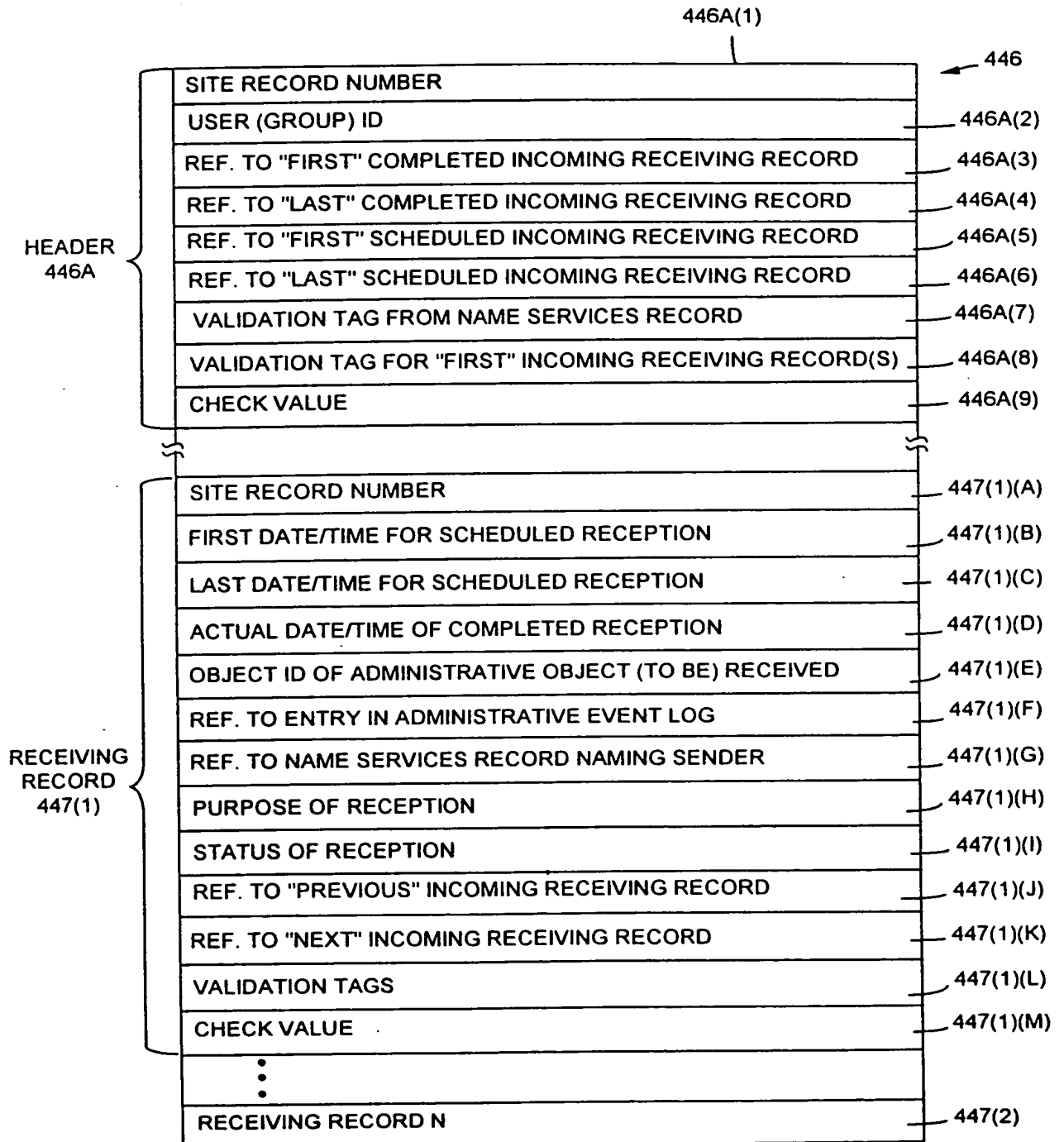


FIG. 29

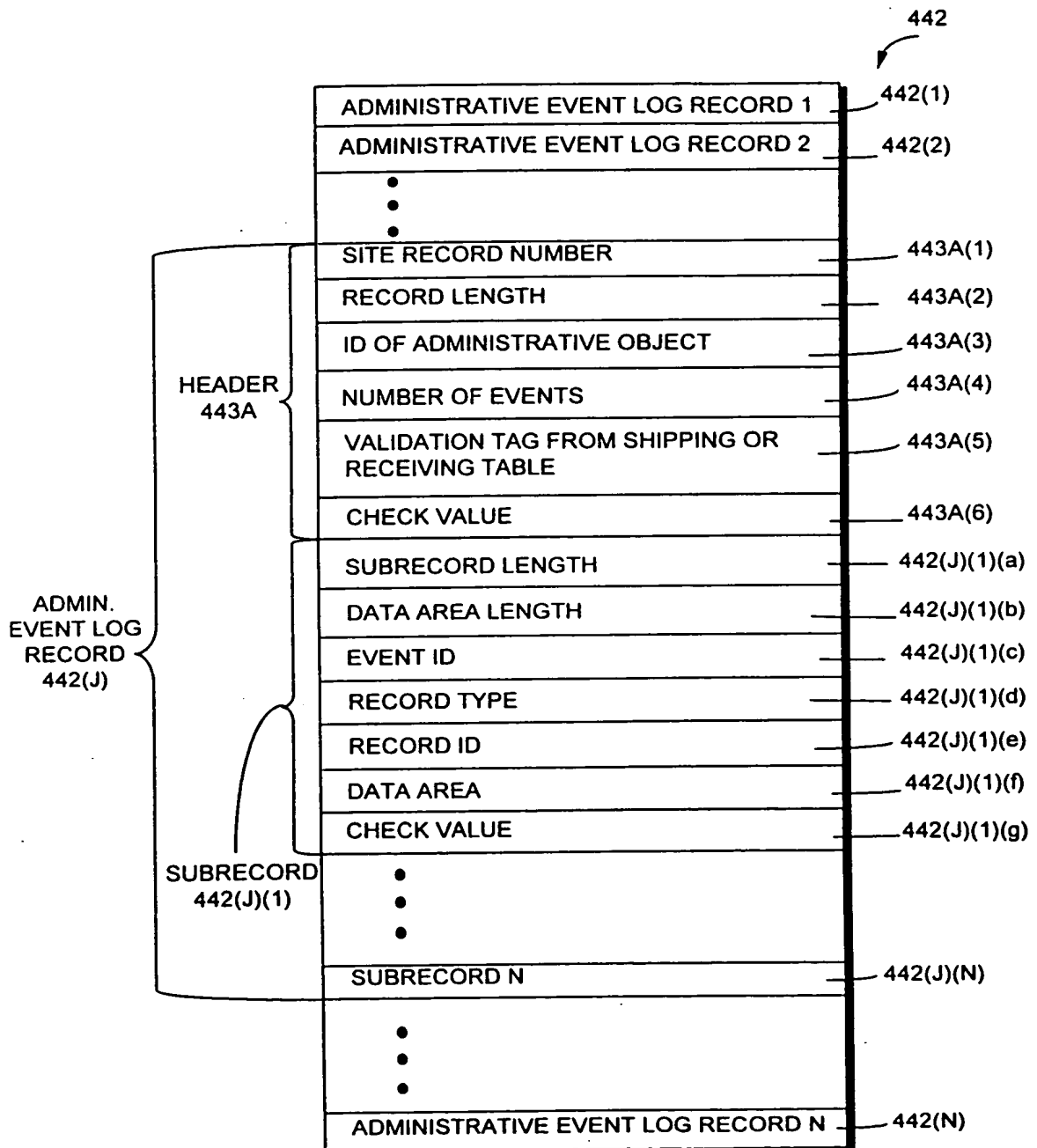
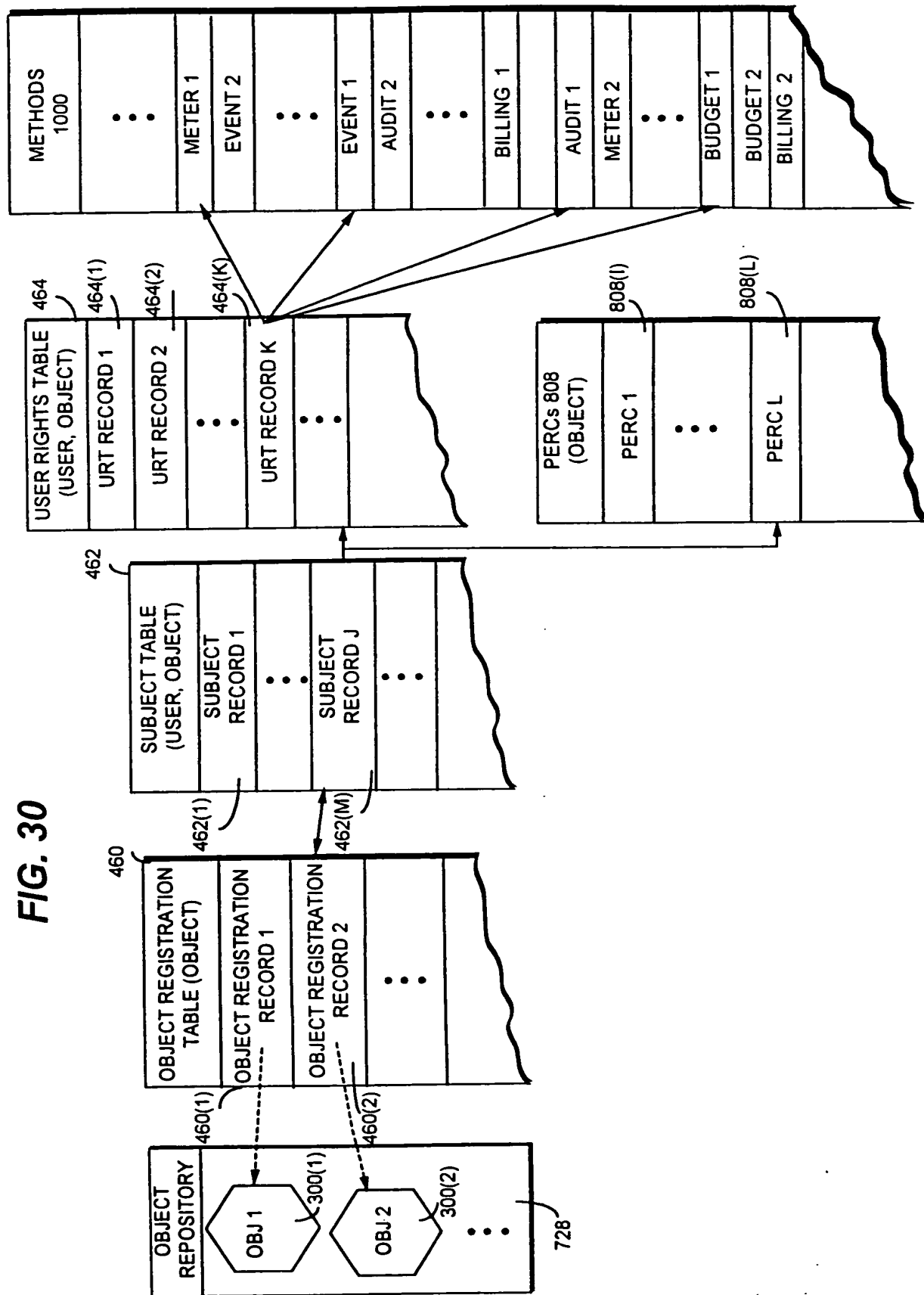


FIG. 30



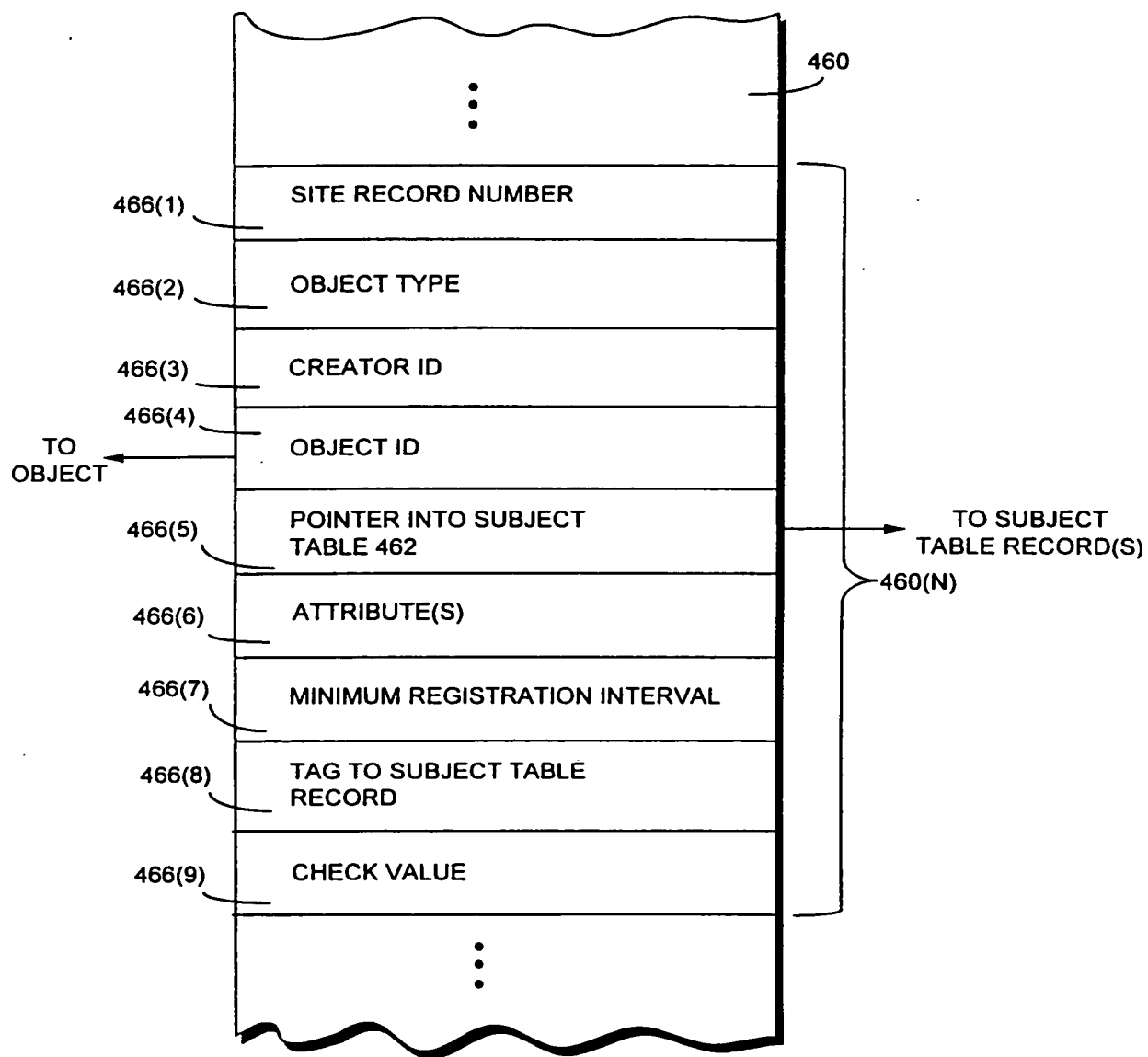


FIG. 31

FIG. 32

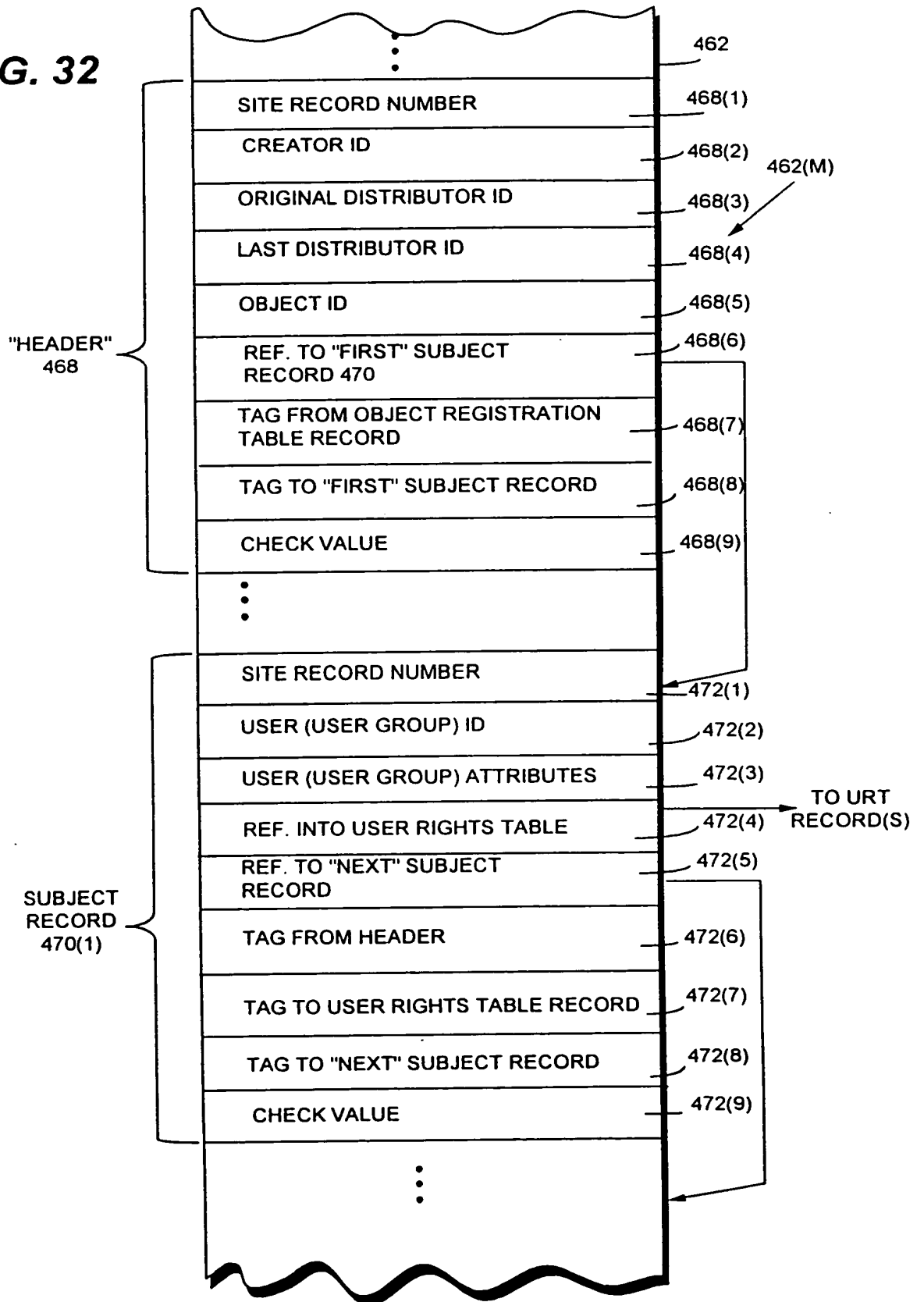


FIG. 33

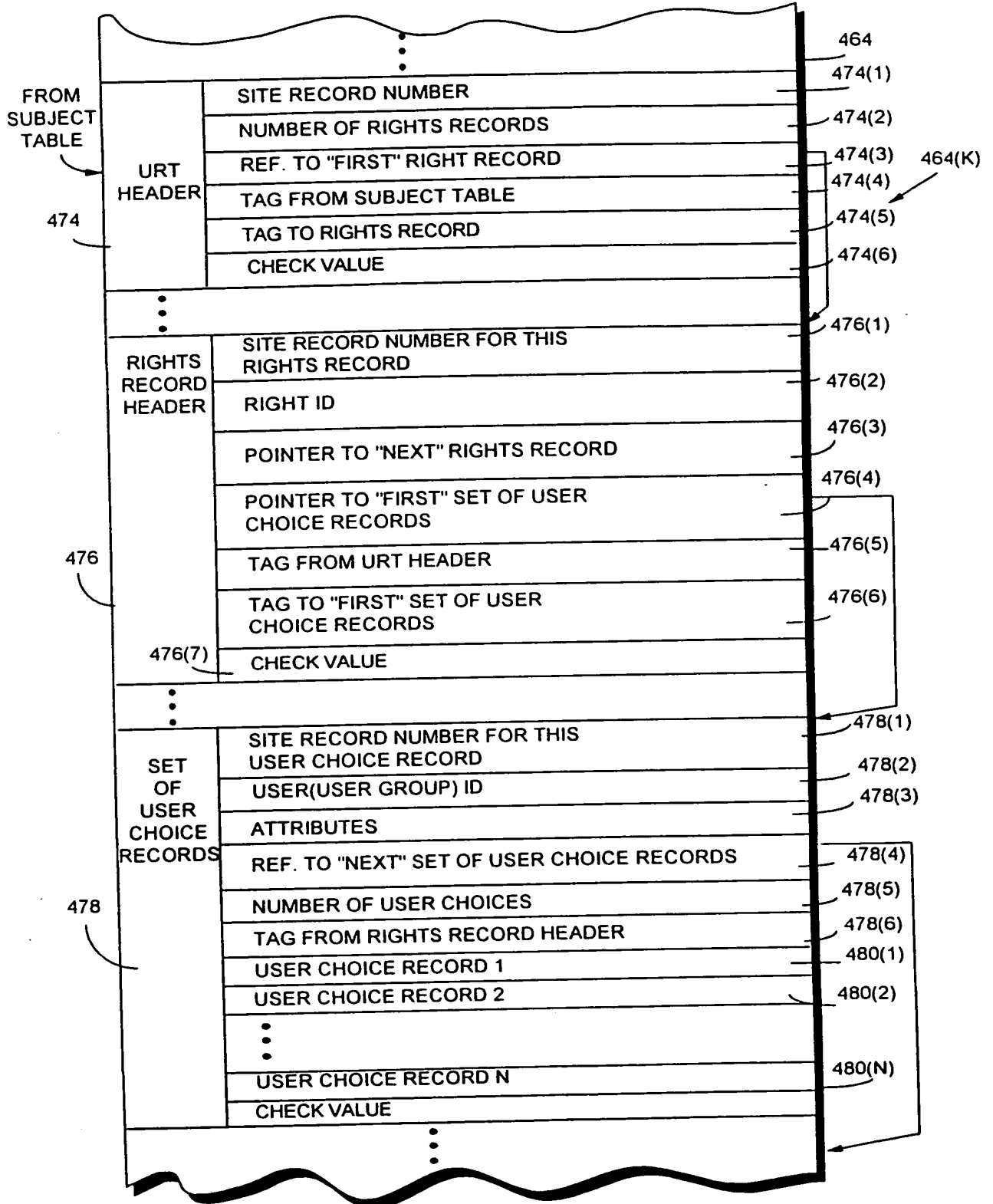


FIG. 34

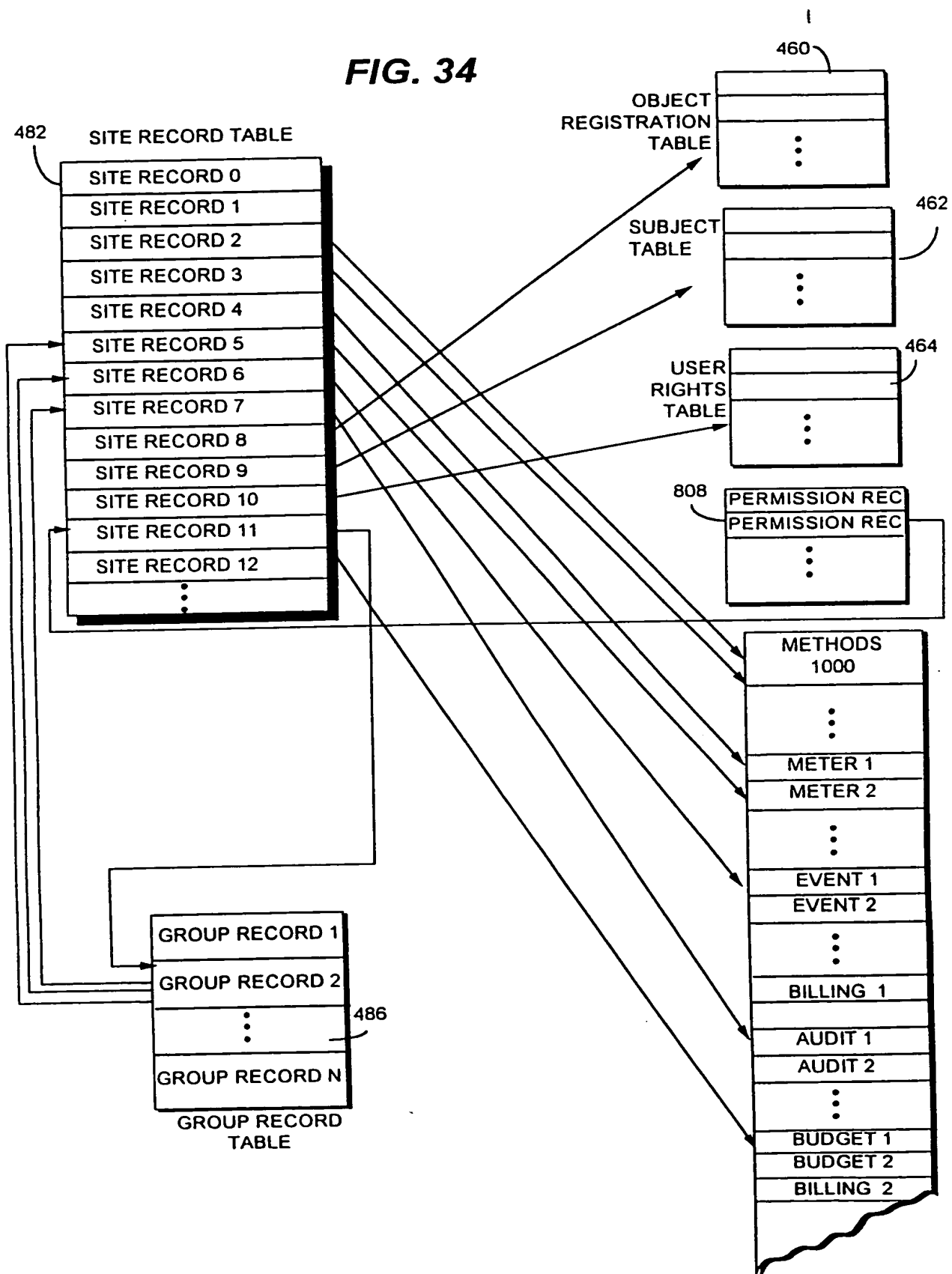


FIG. 34A

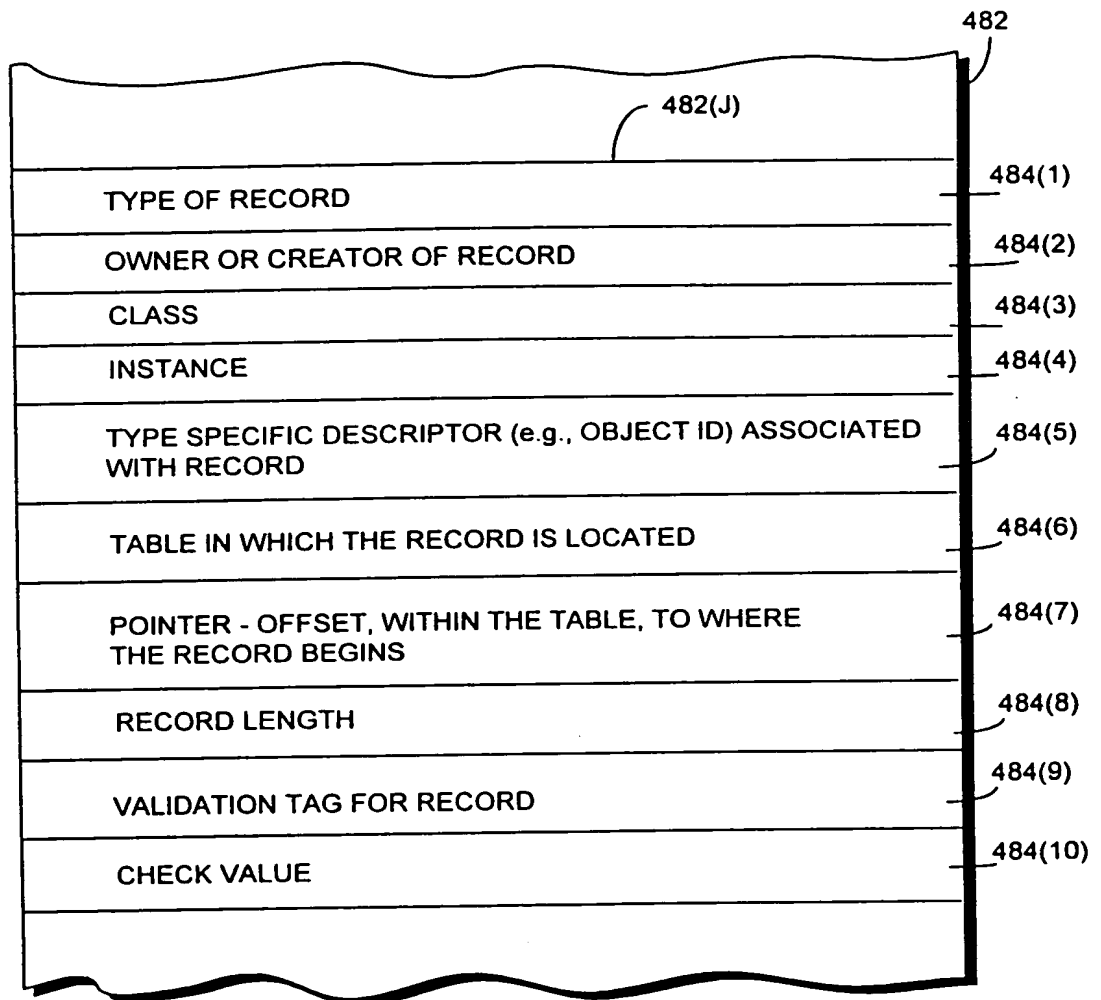


FIG. 34B

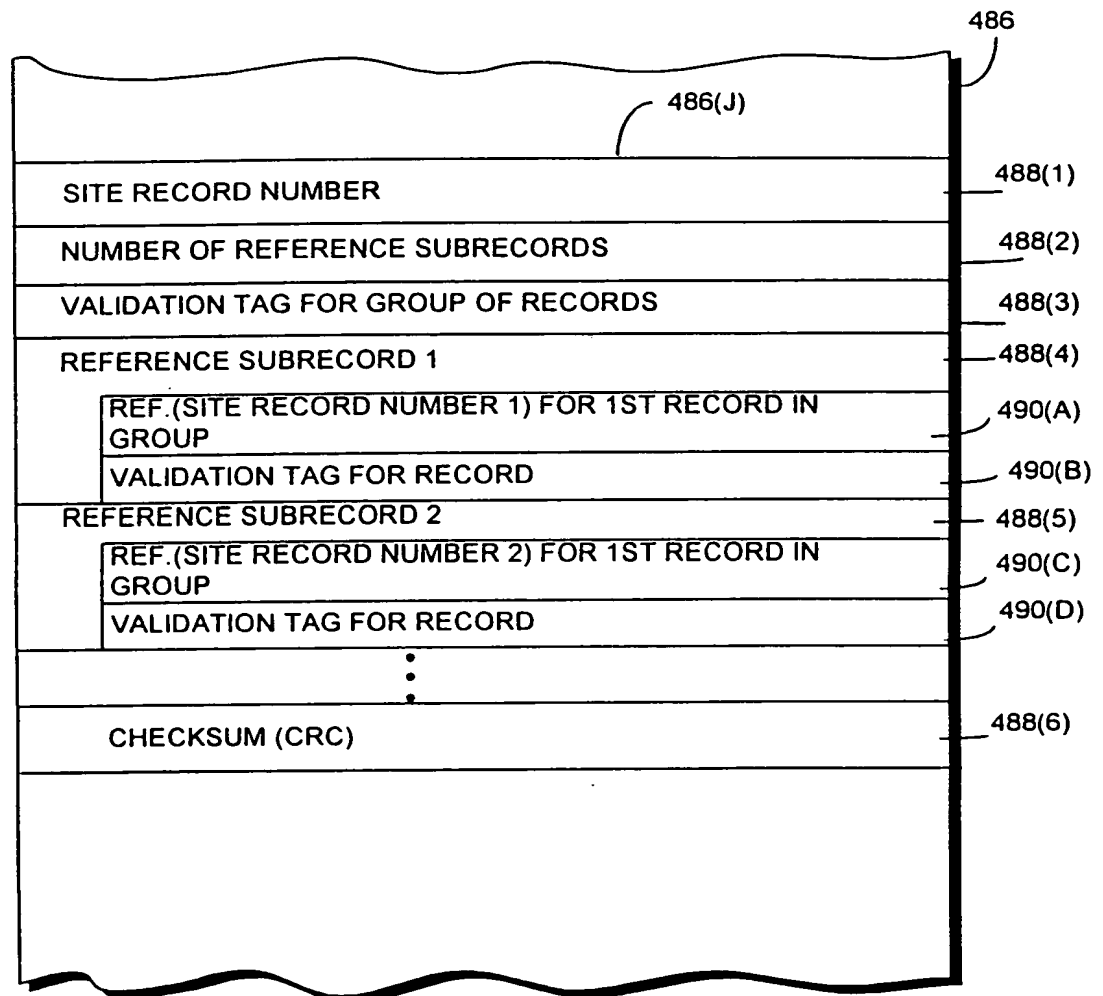
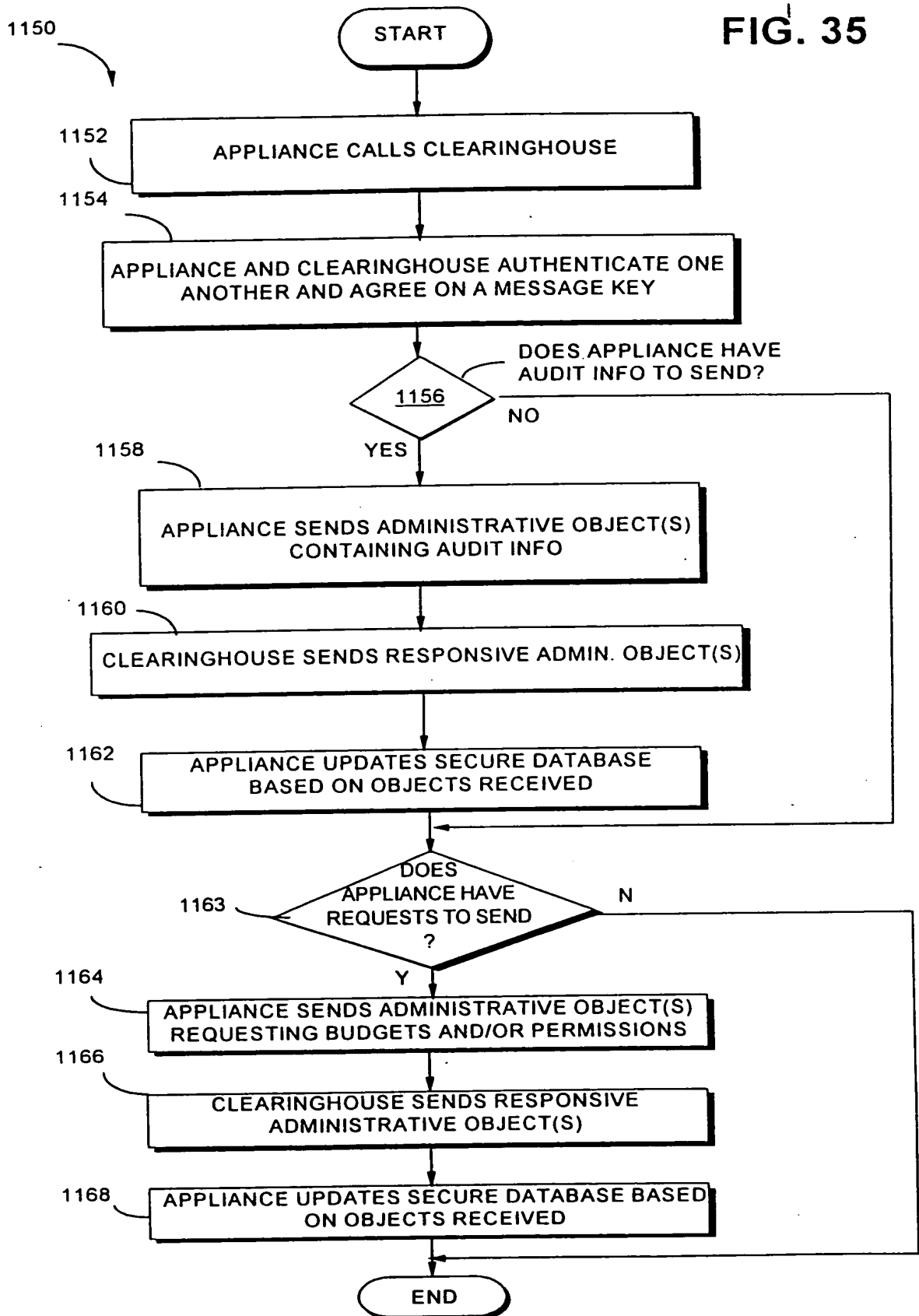


FIG. 35



09070252 100300

FIG. 36

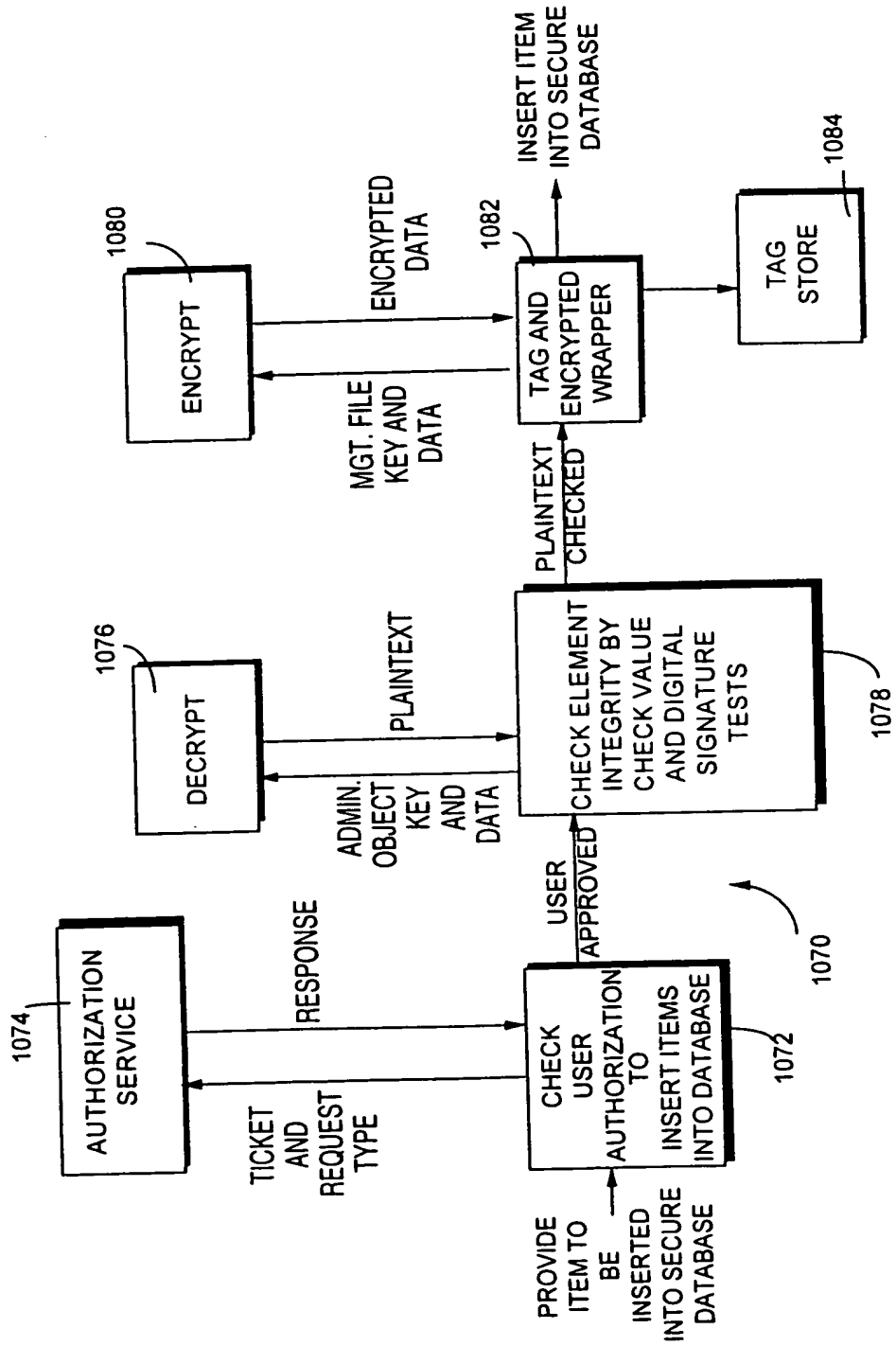


FIG. 37

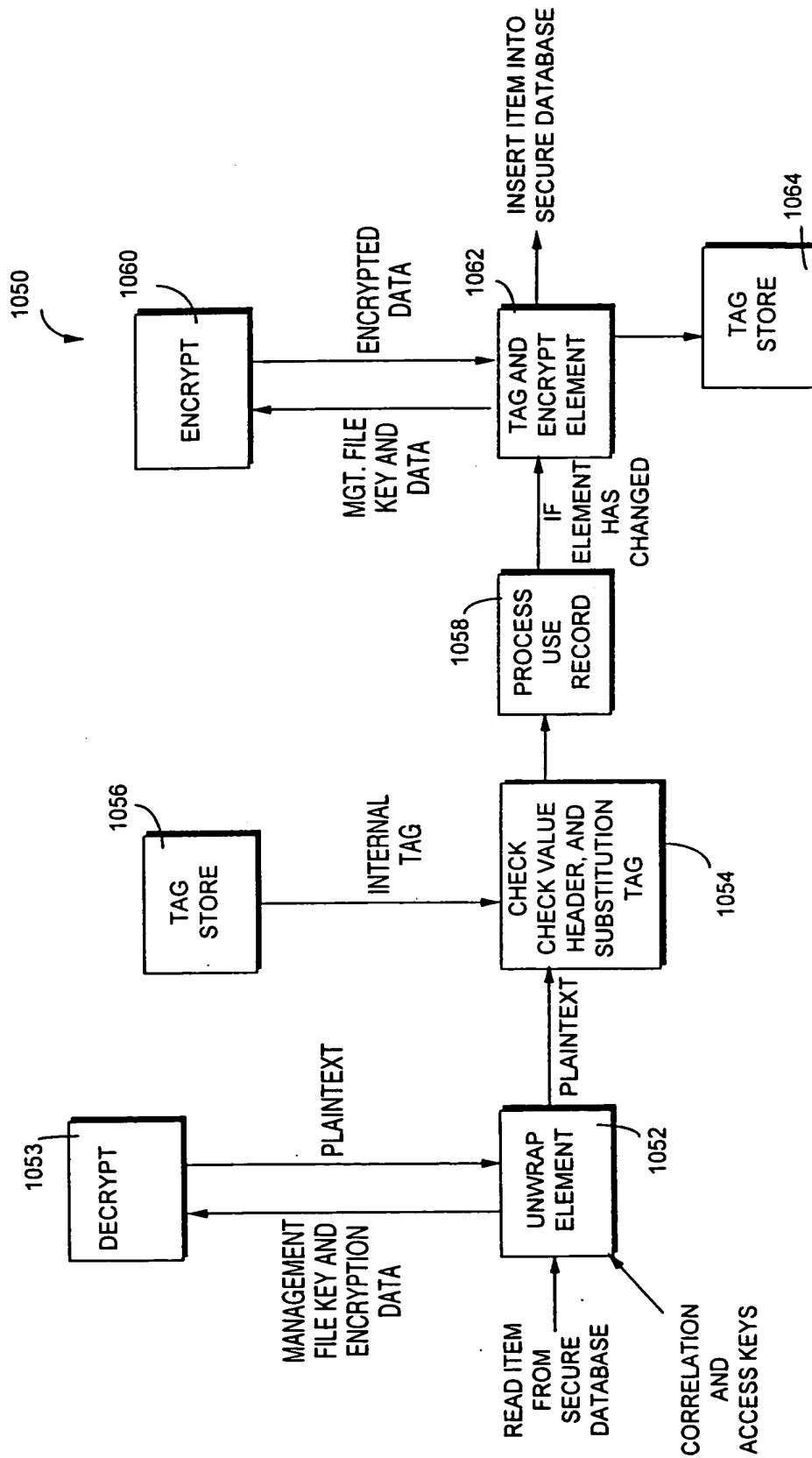


FIG. 38

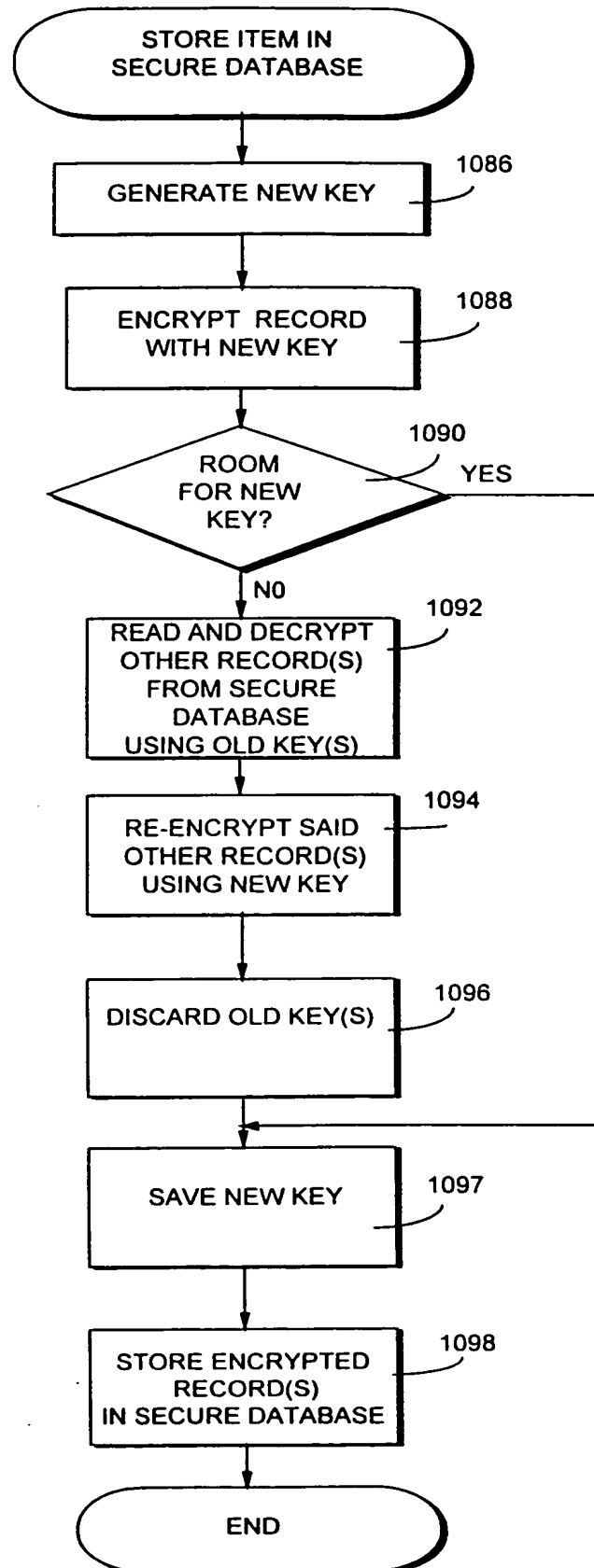


FIG. 39

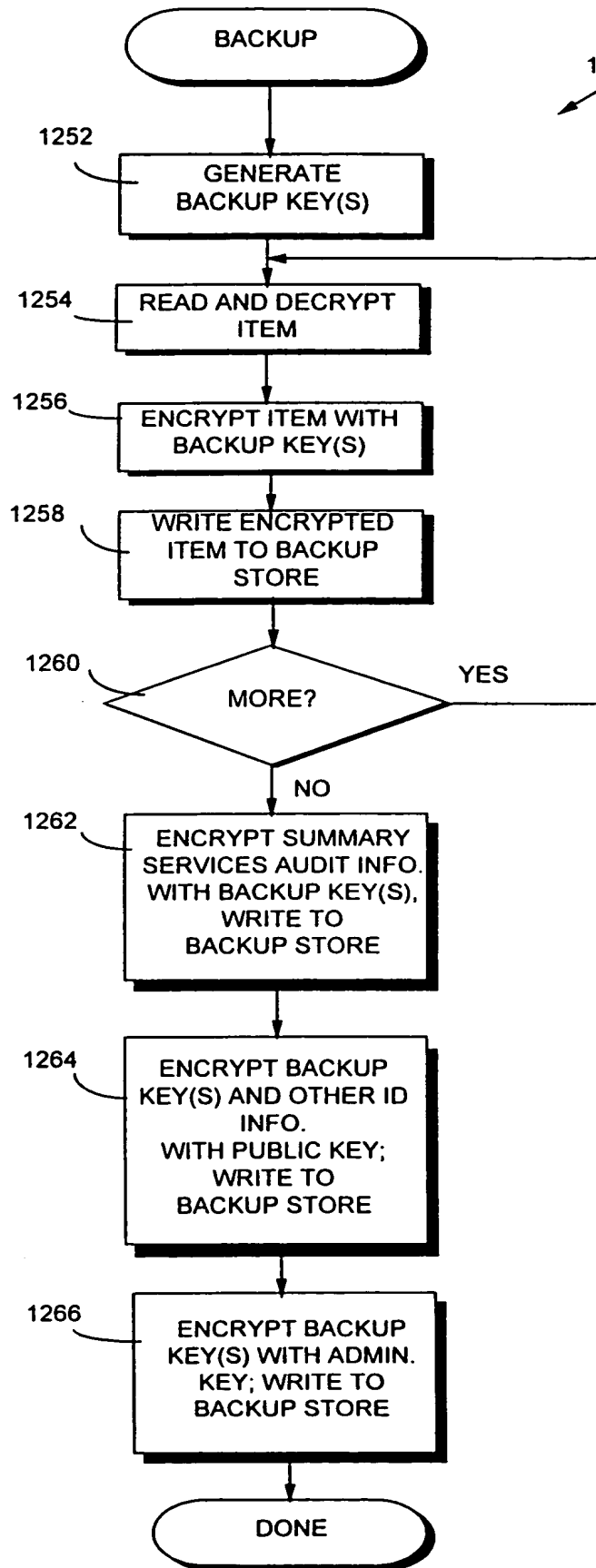
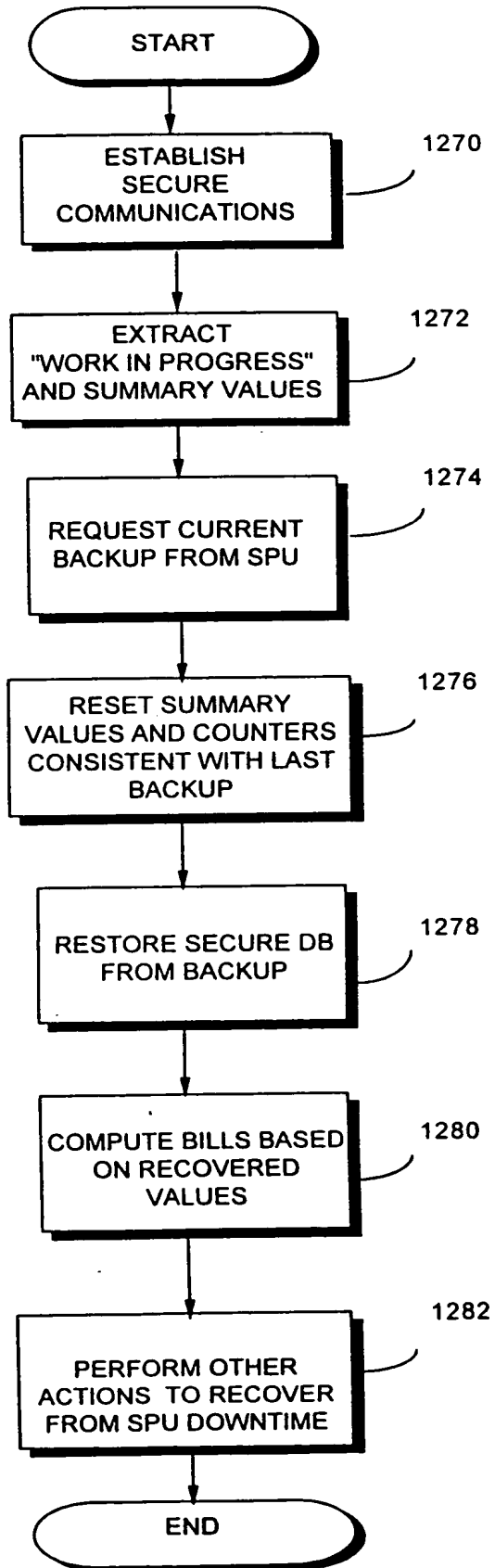


FIG. 40

1268



The diagram illustrates a VDE Node structure. It consists of two main sections, 600A and 600B, each labeled "VDE Node".

Section 600A (bottom) contains a block labeled 1000A, which is divided into a "METHOD" section and a "Request-1" section (1450). The "Request-1" section is further divided into four sub-sections.

Section 600B (top) contains a block labeled 1000B, which is divided into a "METHOD" section and a "Response-1" section (1454). The "Response-1" section is further divided into four sub-sections.

A line labeled 1452, "Event and optional information", connects the "Request-1" section of 600A to the "Response-1" section of 600B.

FIG. 41a

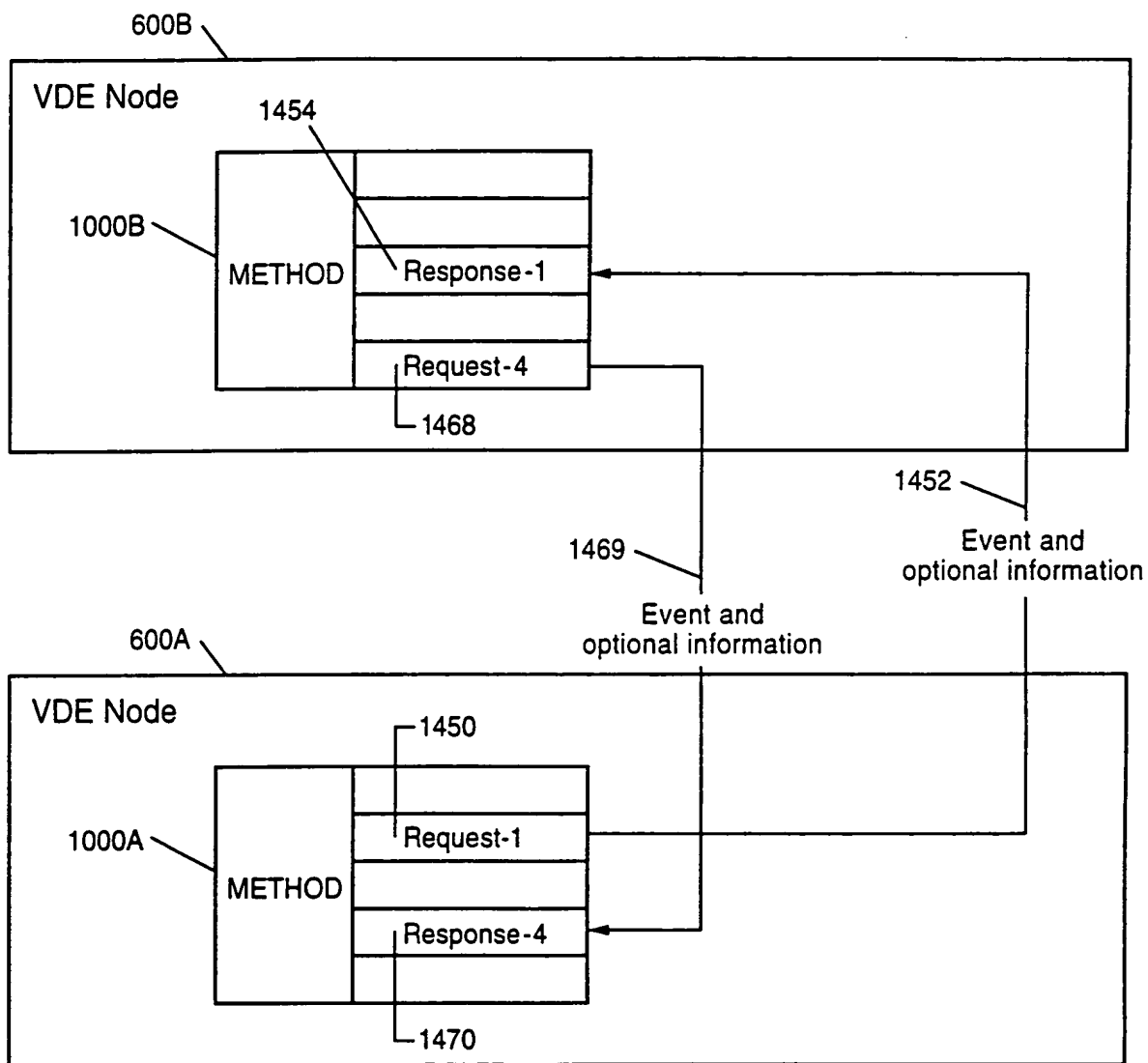


FIG. 41b

09676252 100500

FIG. 41c

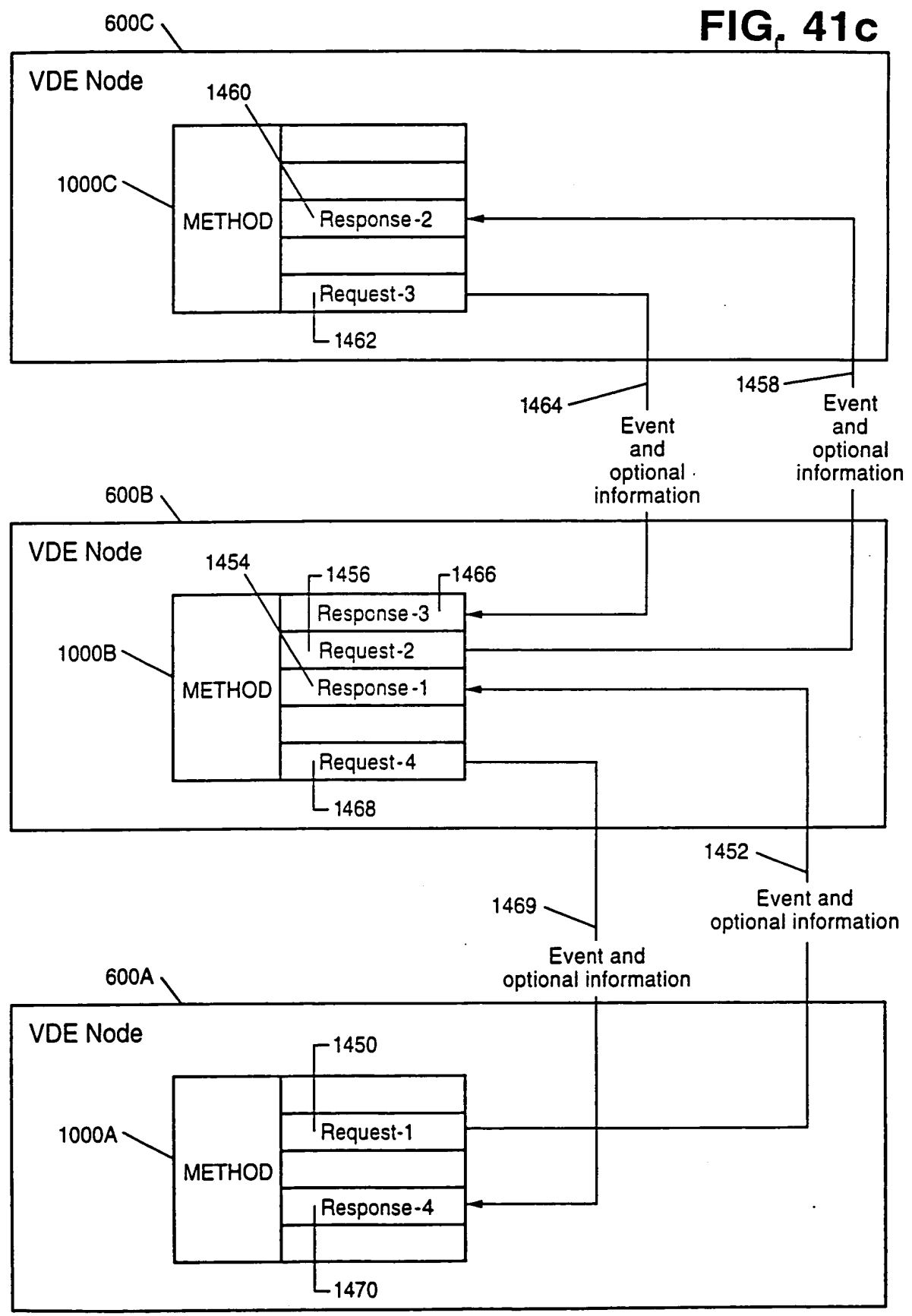
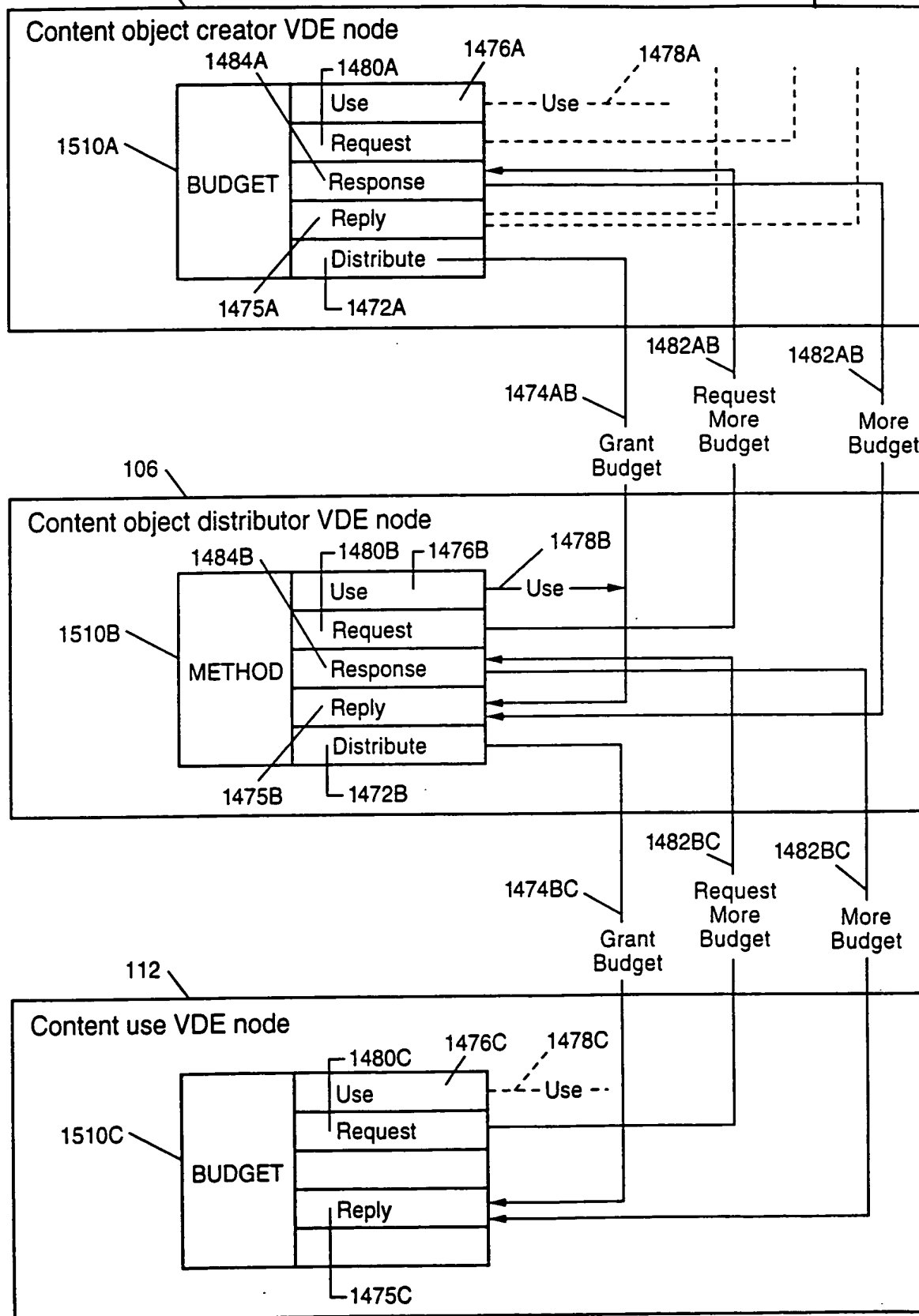


FIG. 41d



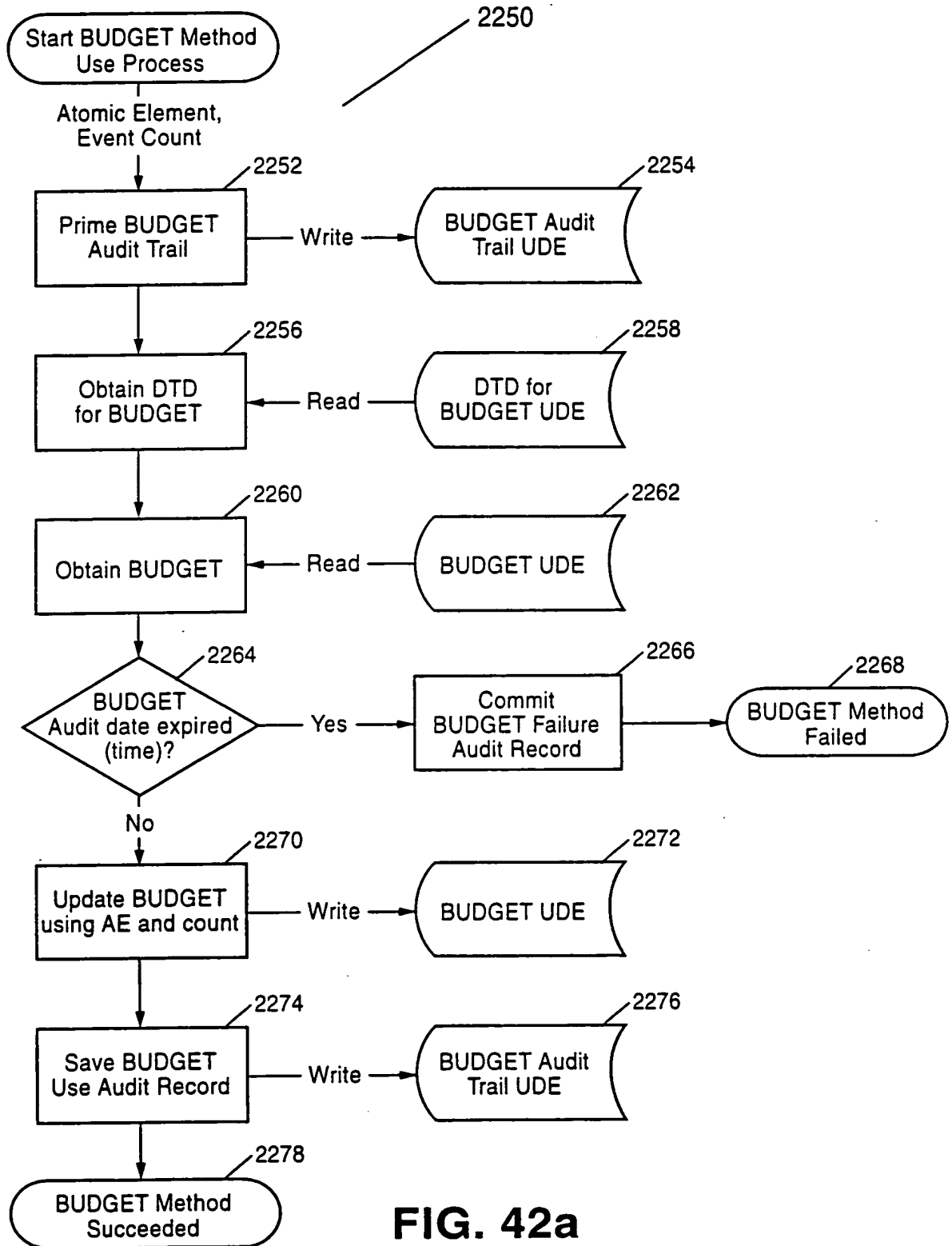


FIG. 42a

09070350 100300

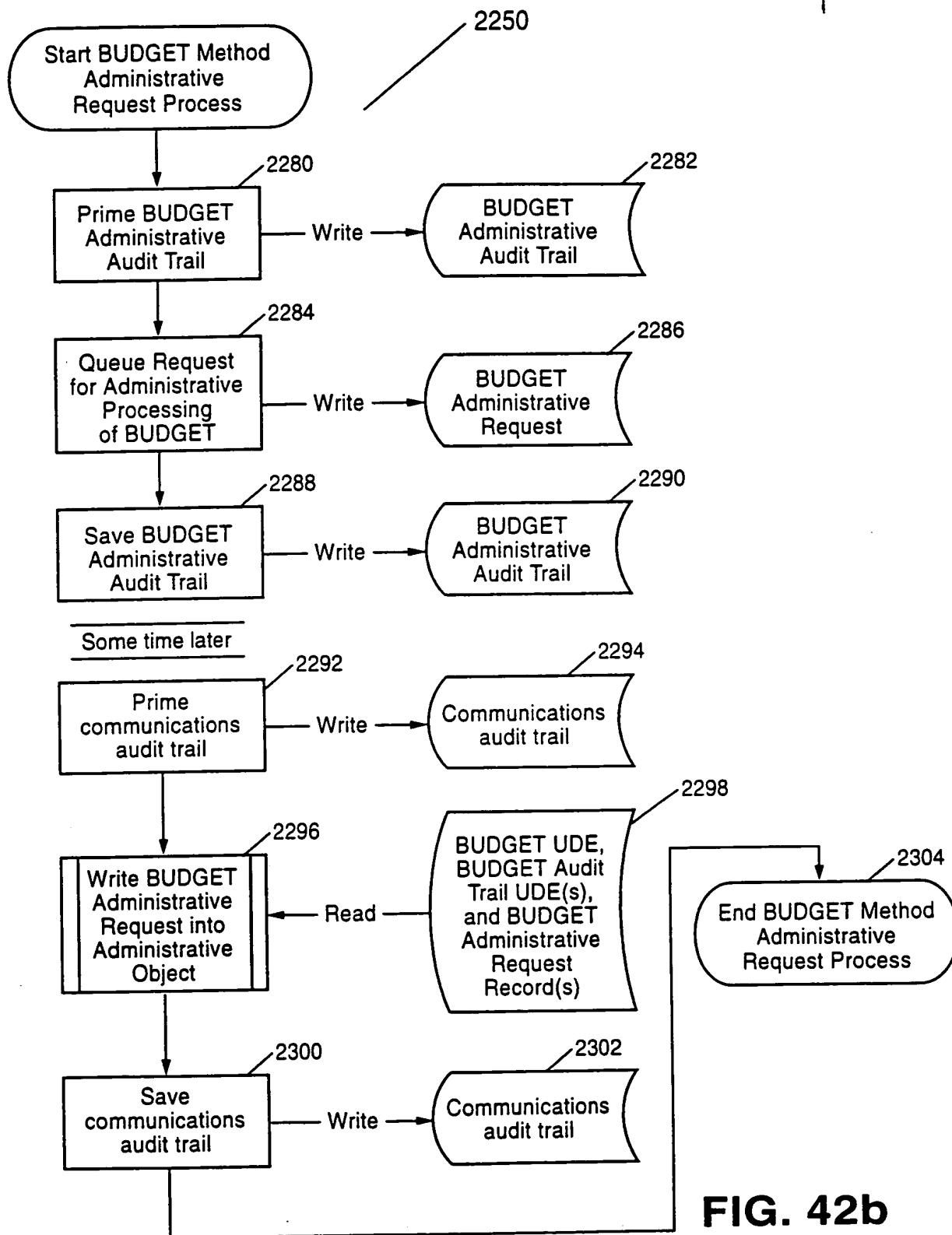


FIG. 42b

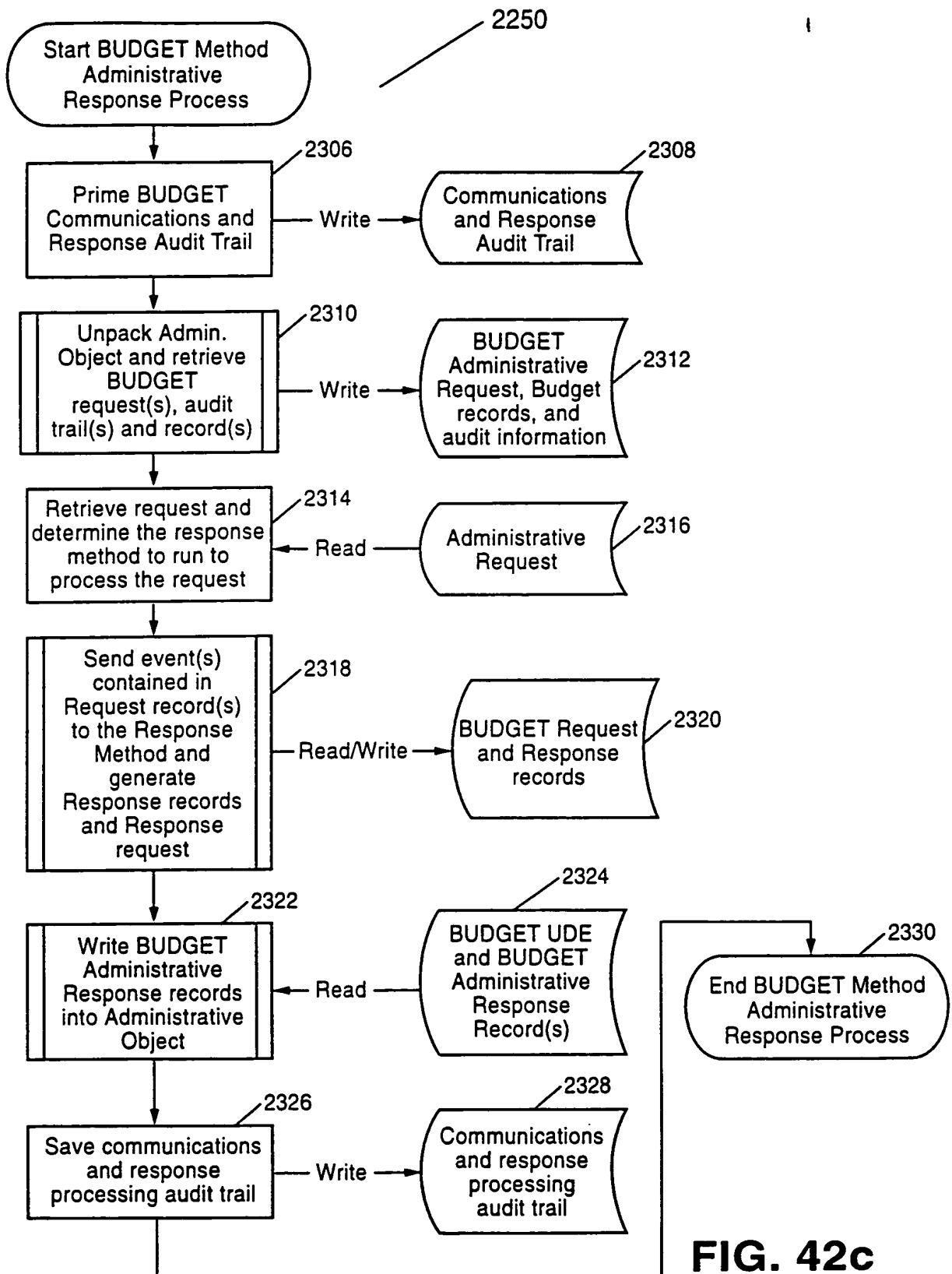


FIG. 42c

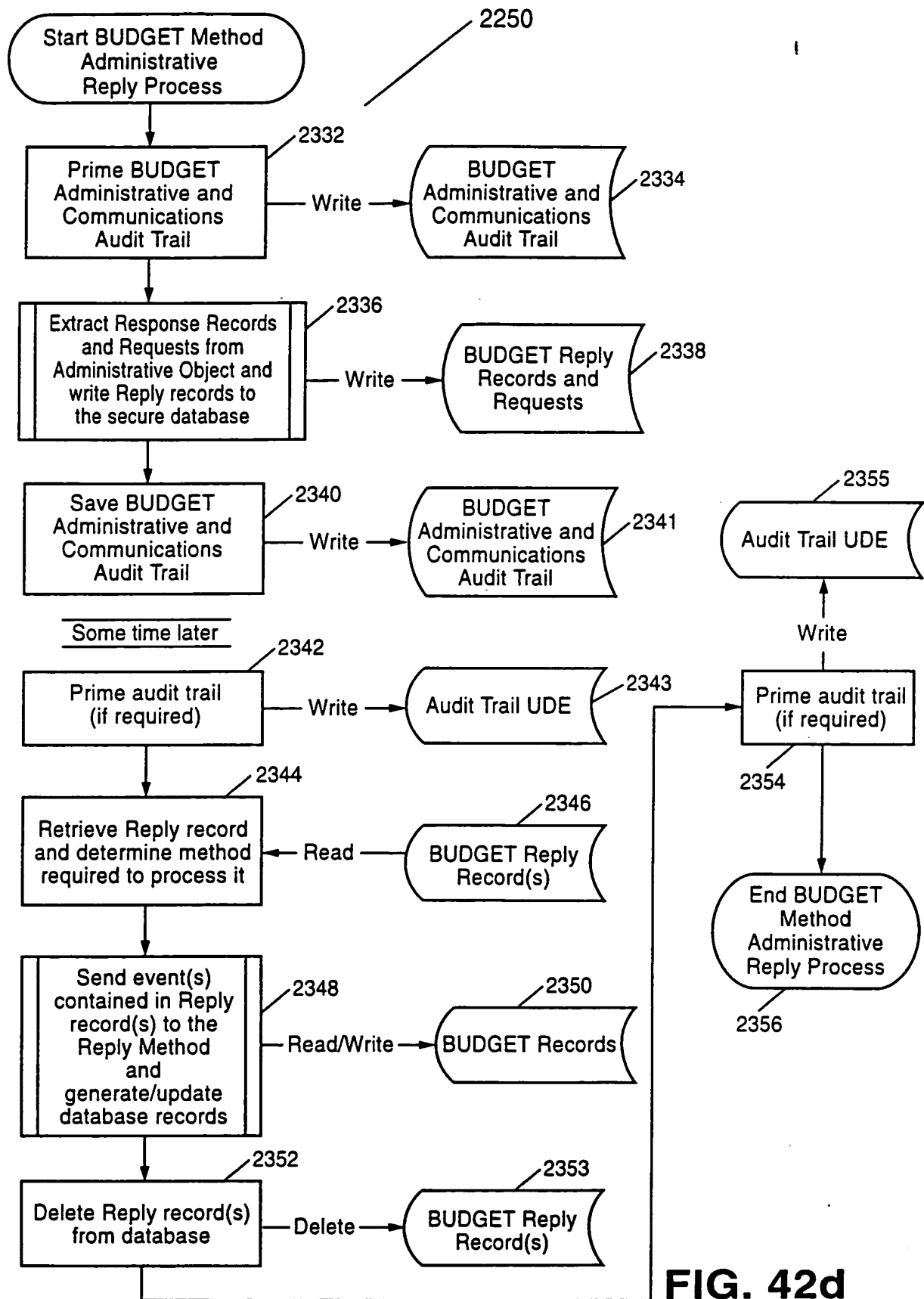


FIG. 42d

FIG. 43a

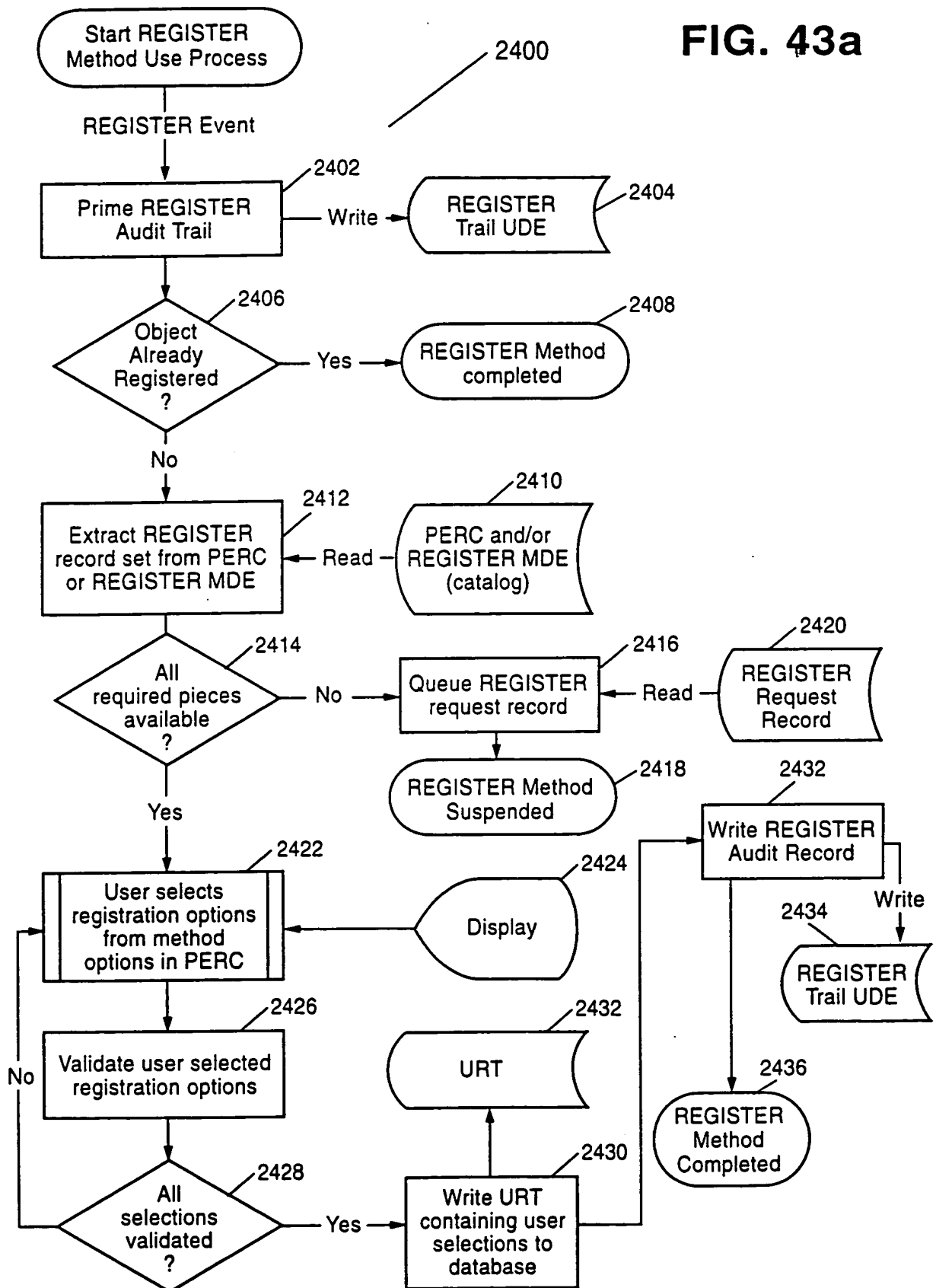


FIG. 43b

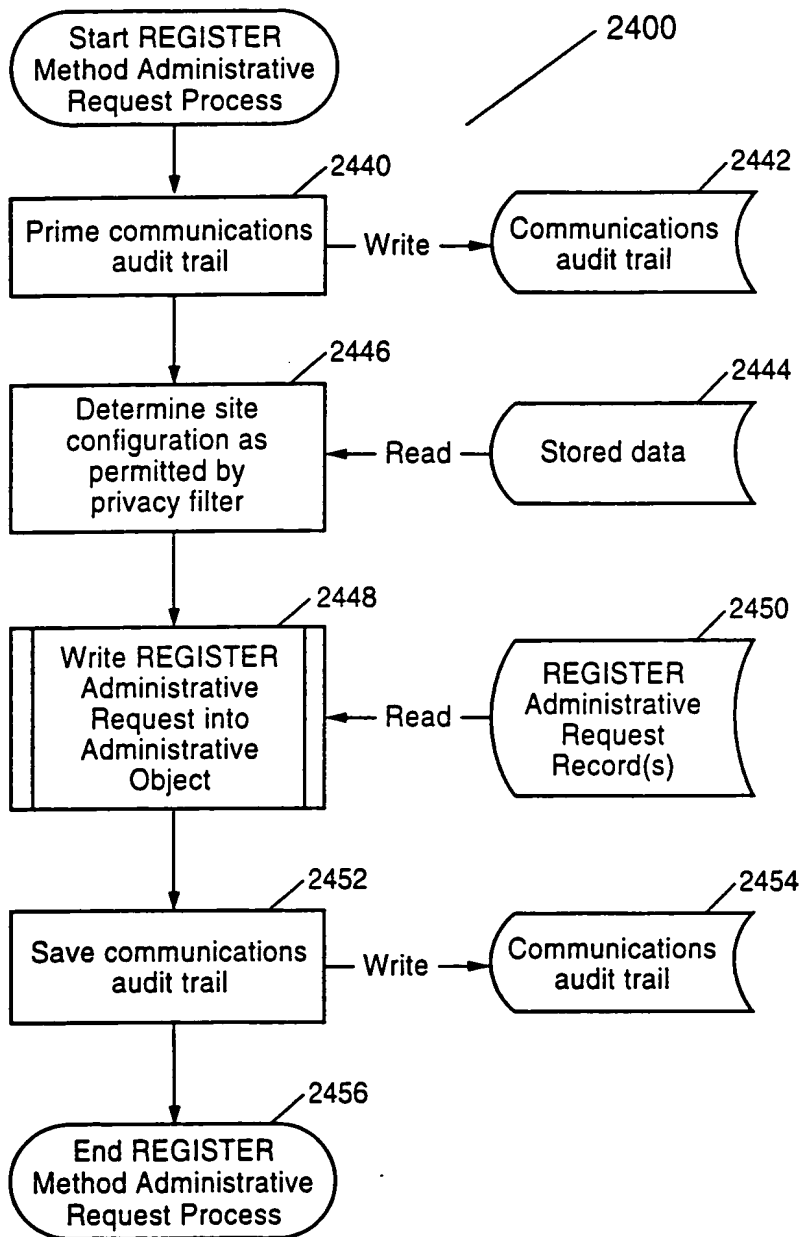


FIG. 43c

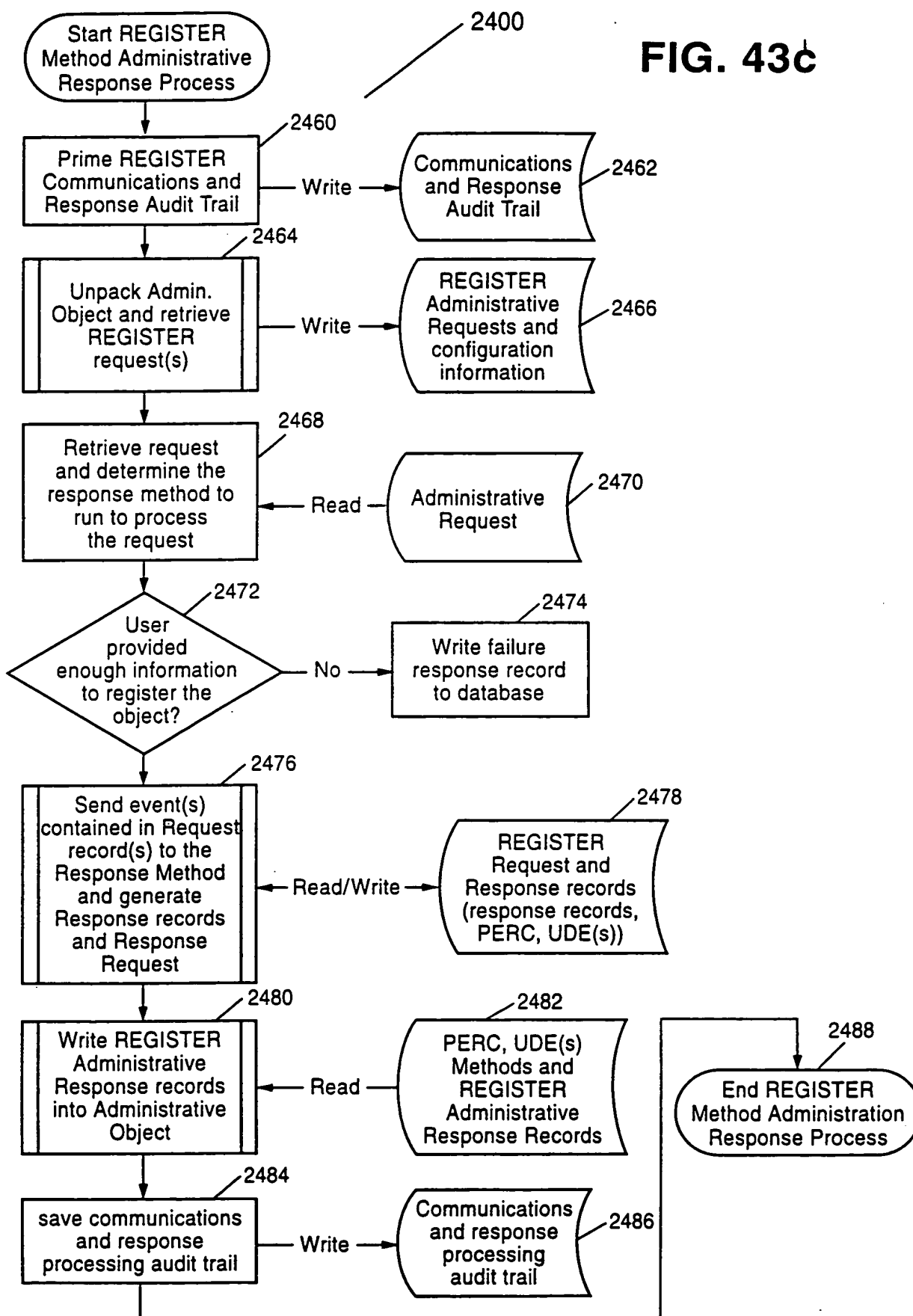




FIG. 44a

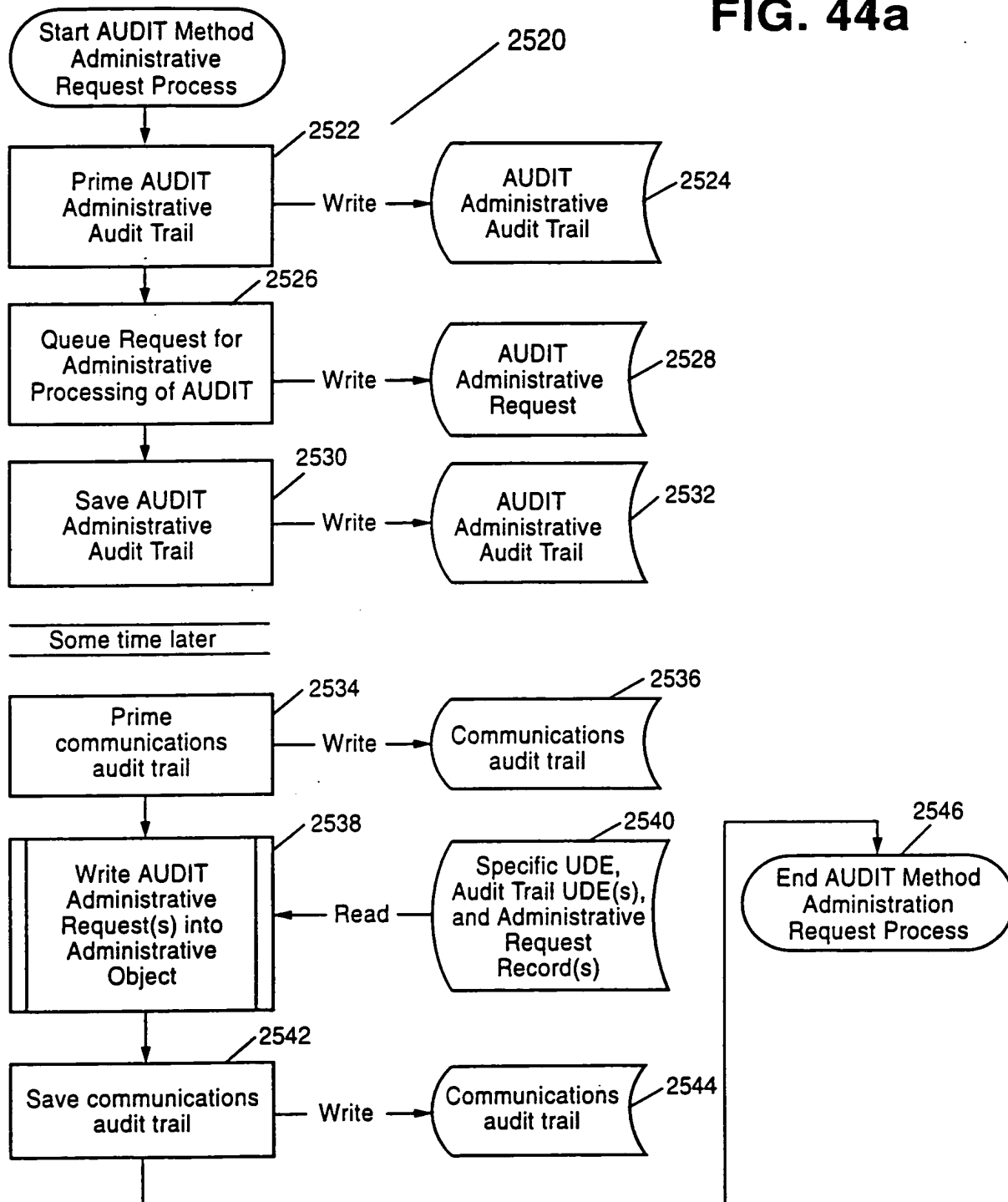


FIG. 44b

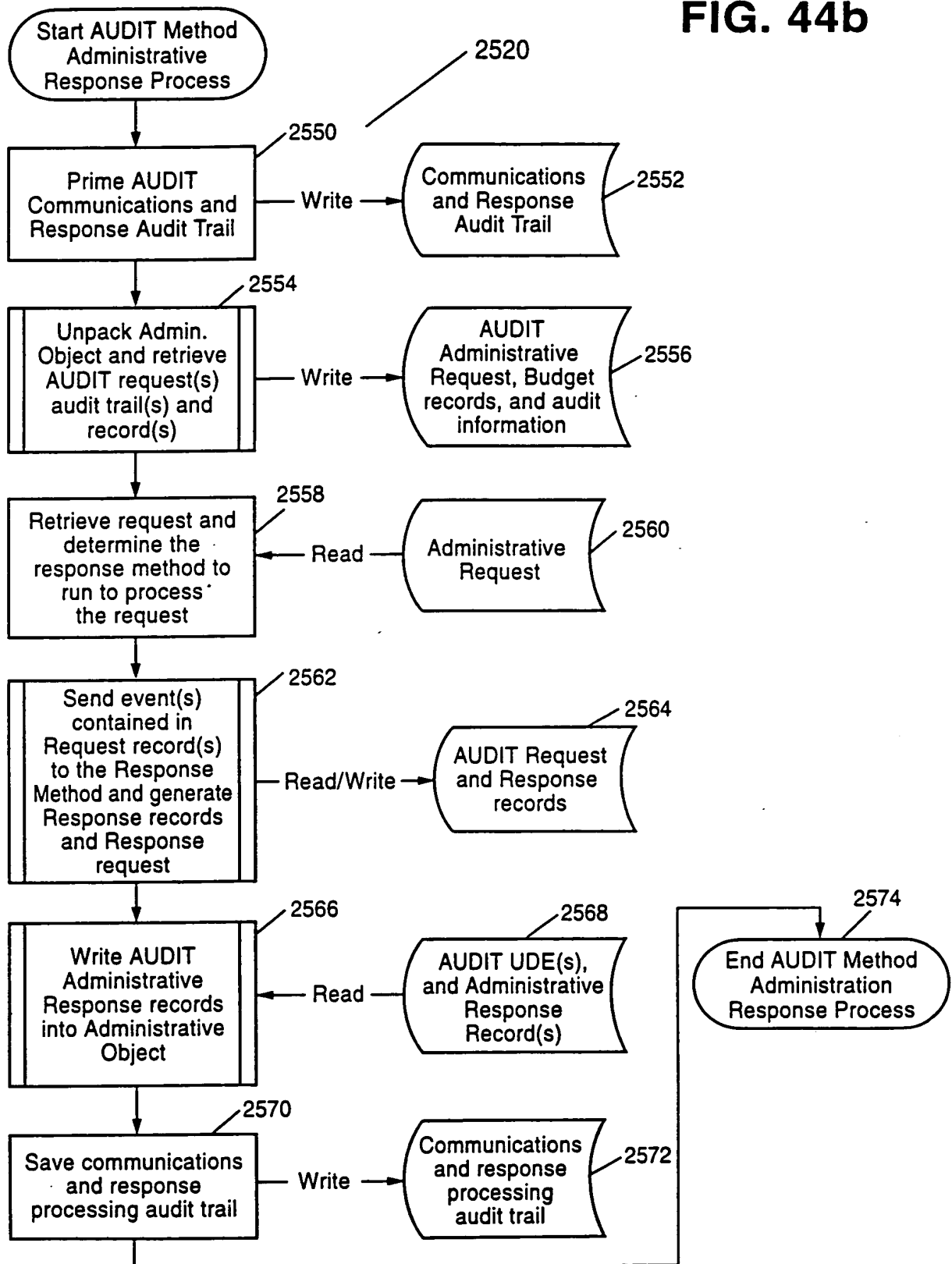


FIG. 44c

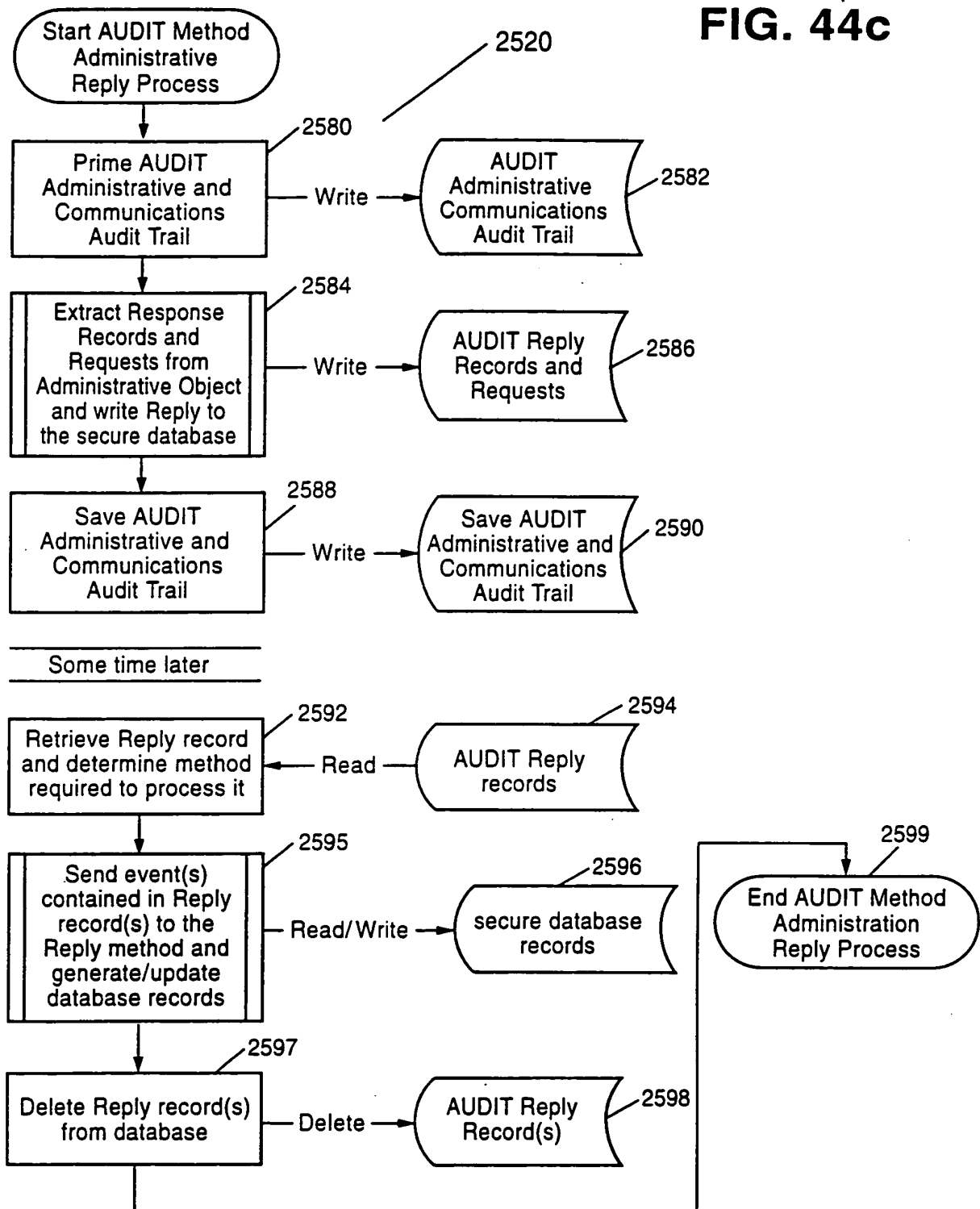


FIG. 45

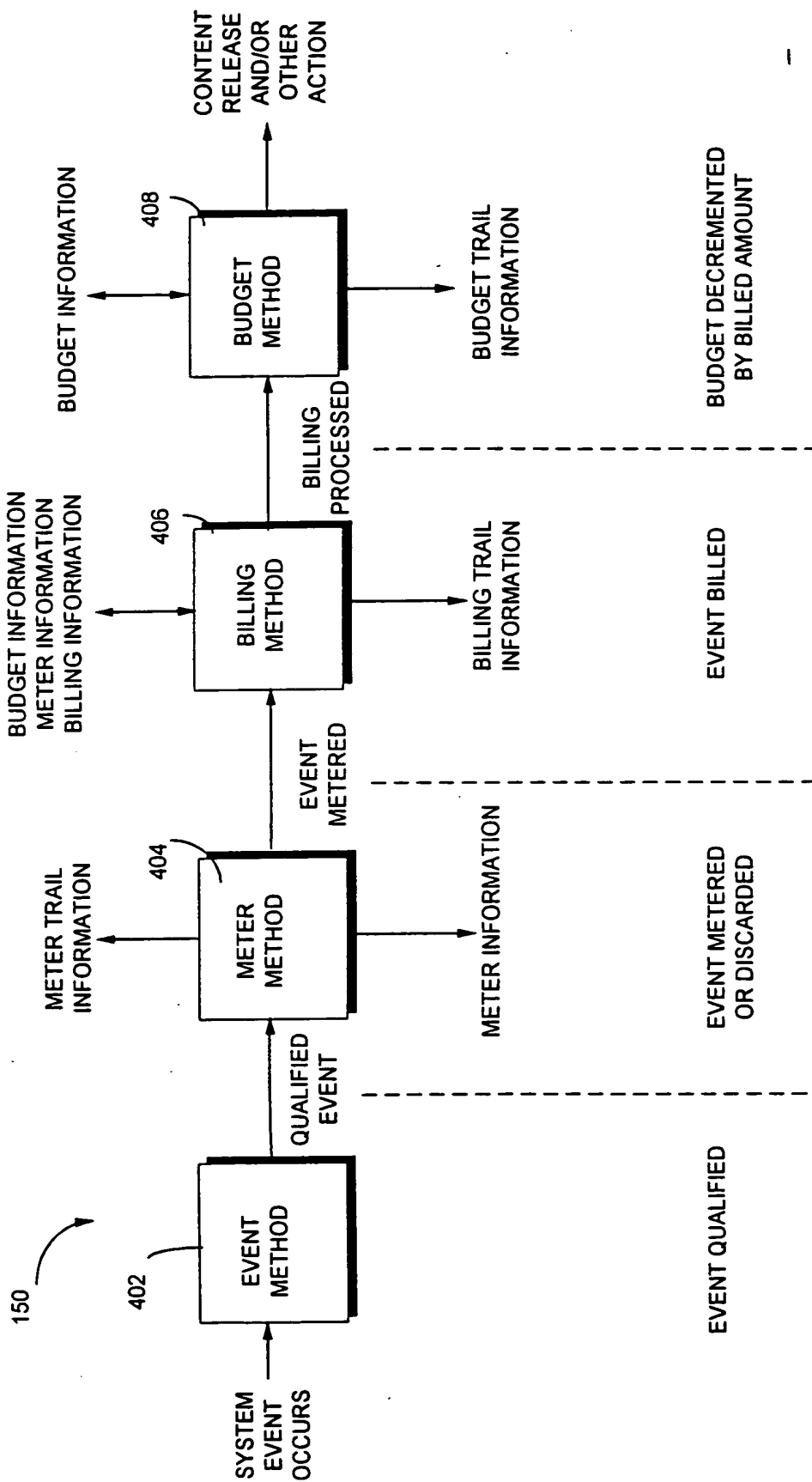
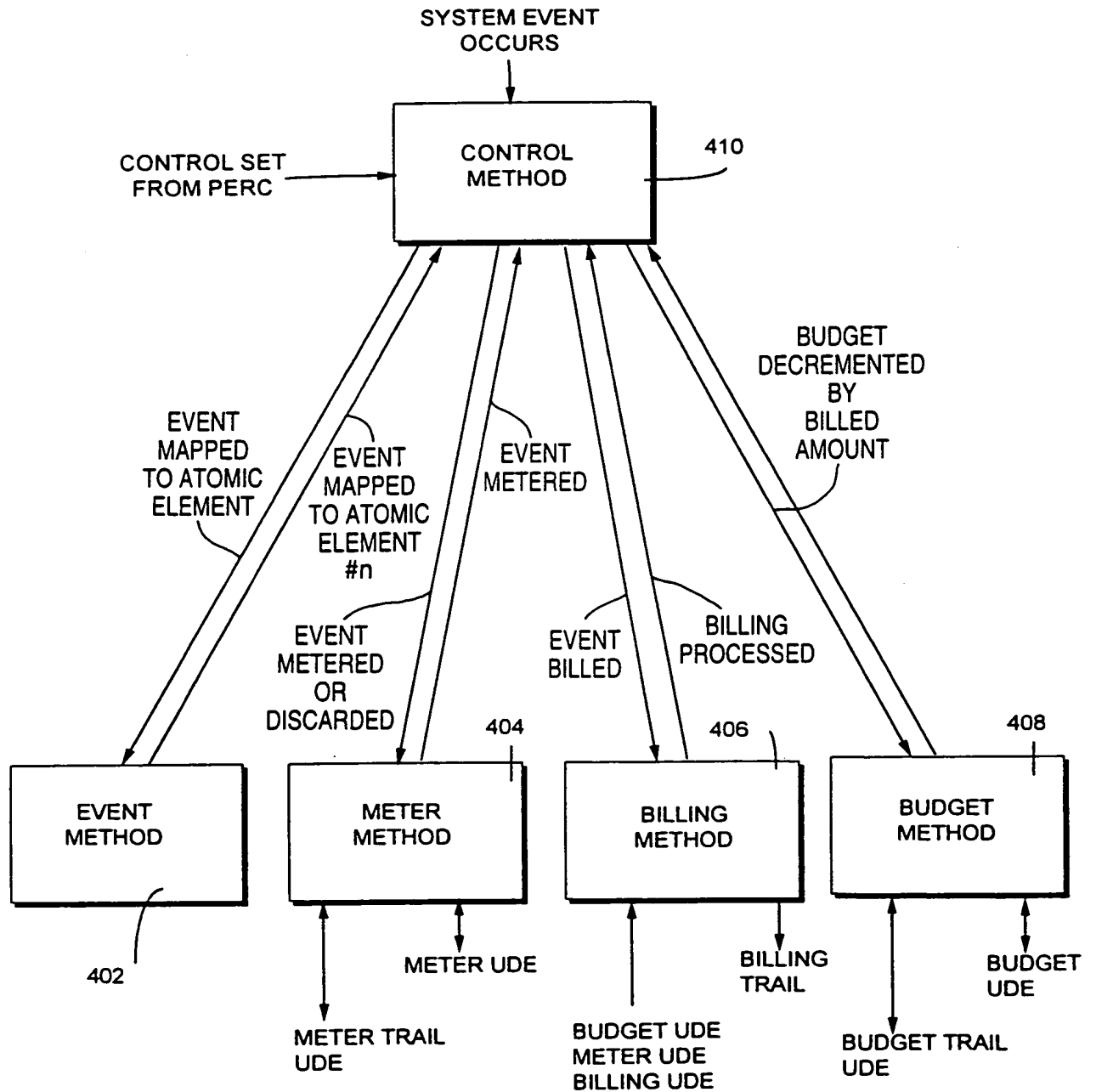
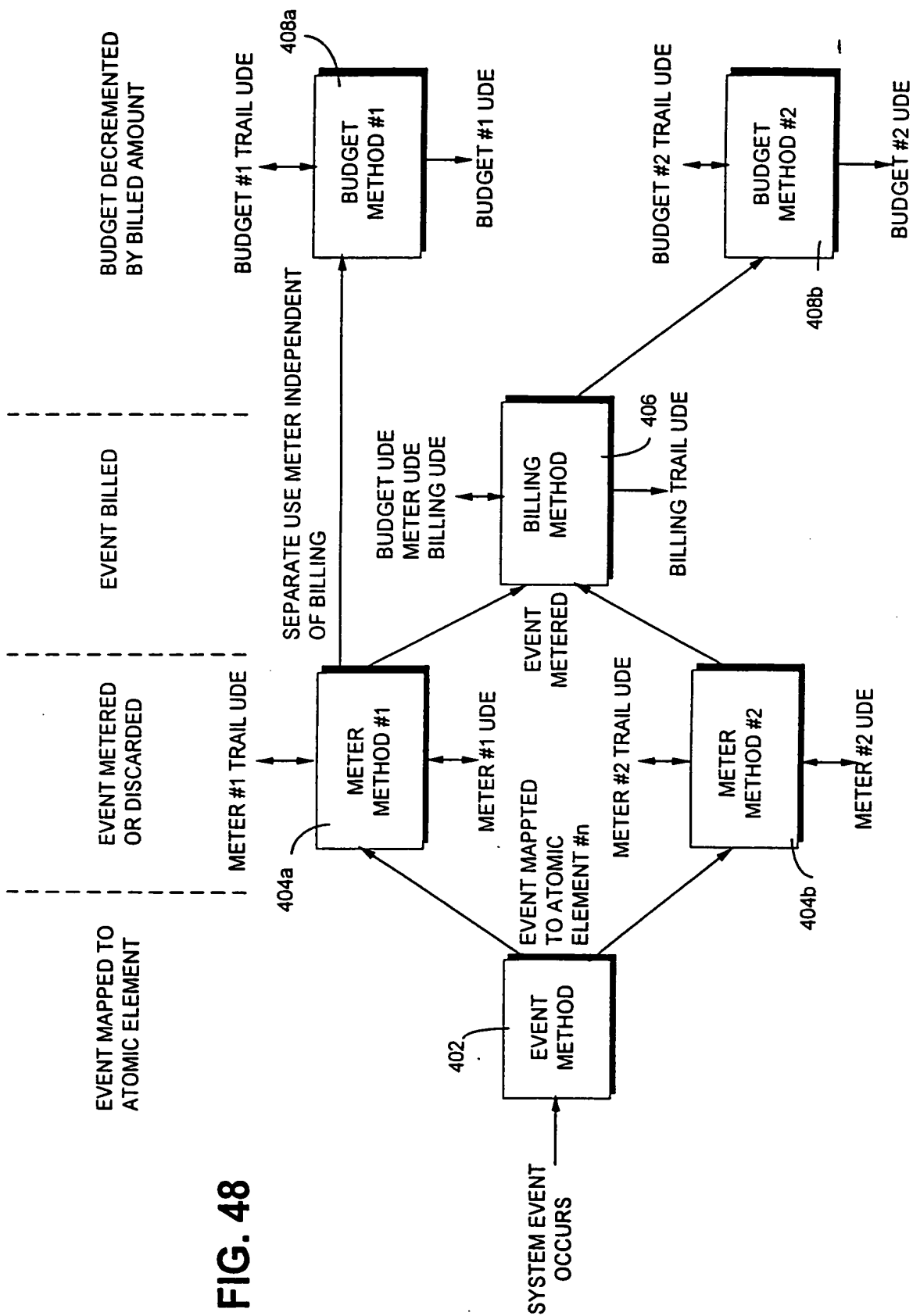


FIG. 46

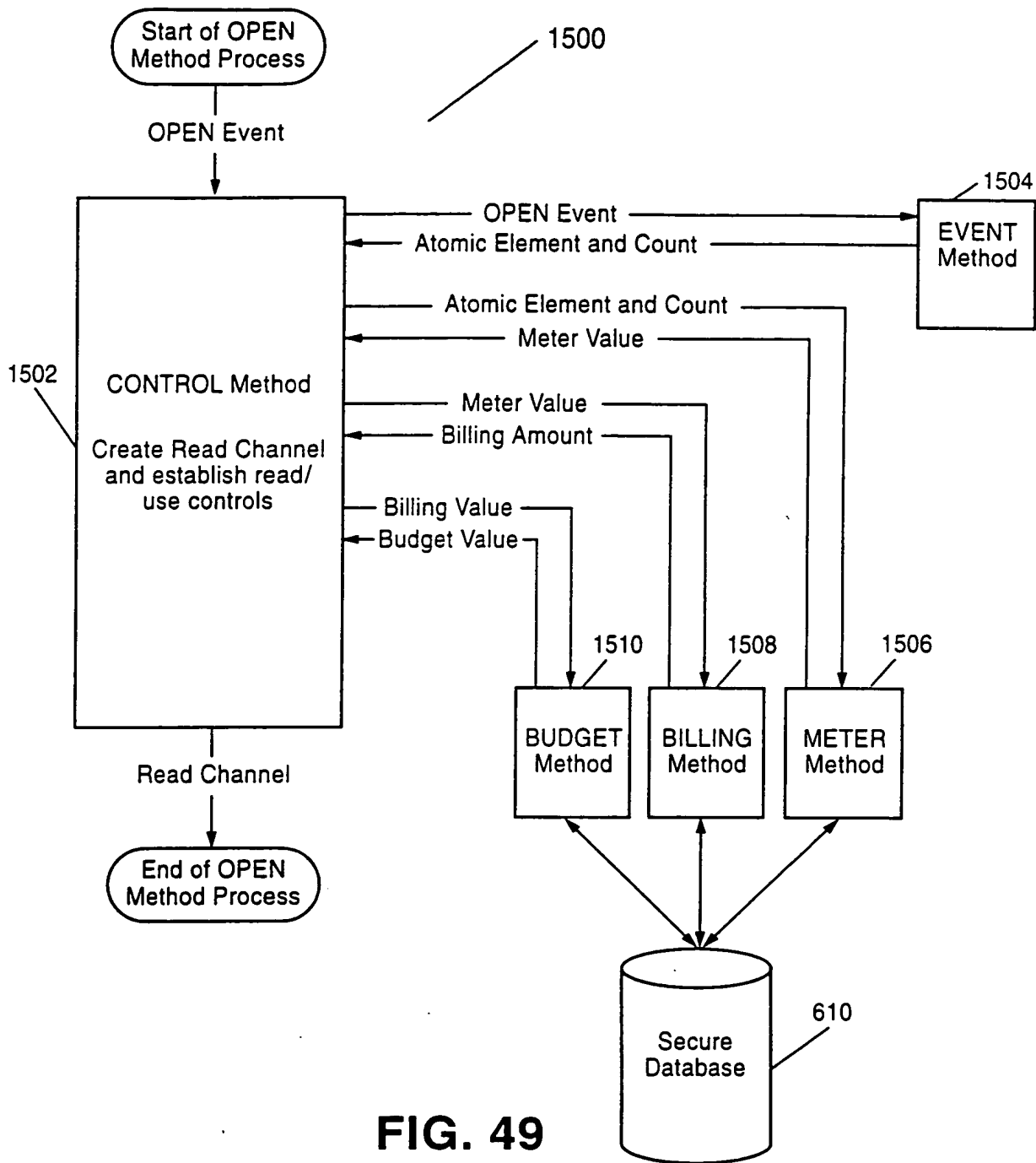


09676252 400300

FIG. 48



00070250 10000



1500

Start of OPEN
Method Process

1502

Open Event

Determine
identification
of object and user
to be opened.

1520

Open Event, Object ID, User ID

URT, PERC for
(object, user)

1524

Read

Is
the Object
registered for this
user?

1522

No

Call the
REGISTER
Method for
the Object.
Restart the
OPEN Method
once the
registration
is complete.

1526

Yes

Is
the Object
already open for
this user?

1528

No

OPEN Method
Elements
(Method core,
LM, UDE, MDE)

1532

Read

Create channel and
bind OPEN control
elements to it

1530

Open Event, Object ID, User ID, Channel ID

Audit UDE

1534

Write

Prime Audit
(if required)

1533

Start Secure
Database Transaction

1536

CONTROL Method

A

FIG. 49a

```
graph TD
    A((A)) -- 1502 --> 1538{{Prime EVENT Audit Trail (if required)}}
    1538 -- Write --> 1540(EVENT Method Audit Trail UDE)
    1540 -- 1504 --> 1542[Map OPEN Event to Atomic Element # and event count using Map MDE]
    1544(EVENT Method Map MDE) -- Read --> 1542
    1542 -- "Event, Event Count, Atomic Element #, Object ID, User ID" --> 1546[Write EVENT Audit Trail (if required)]
    1546 -- Write --> 1548(EVENT Method Audit Trail UDE)
    1548 -- 1504 --> 1550{Atomic Element Selected?}
    1550 -- "Yes, Pass" --> 1552{EVENT Method Succeeded?}
    1550 -- "No, Fail EVENT Method" --> 1554[Roll back secure database transaction]
    1552 -- Yes --> B((B))
    1552 -- No --> 1554
    1554 --> 1556([OPEN Method Failed])
    1556 -- "CONTROL Method (cont'd)" --> B
```

FIG. 49b

```

graph TD
    B((B)) -- 1502 --> 1558{{Prime METER Audit Trail (if required)}}
    1558 -- Write --> 1560[METER Method Audit Trail UDE]
    1558 --> 1564[Add EVENT Count to Meter Value]
    1564 <--> |Read/Write| 1562[METER Method UDE (the Meter)]
    1564 --> 1566[Write METER Audit Trail (if required)]
    1566 -- Write --> 1568[METER Method Audit Trail UDE]
    1566 -- METER Value --> 1570{Meter Increment Succeeded?}
    1570 -- "Yes, Pass" --> 1572{METER Method Succeeded?}
    1570 -- "No, Fail METER Method" --> 1574[Roll back secure database transaction]
    1572 -- Yes --> C((C))
    1572 -- No --> 1574
    1574 --> 1576([OPEN Method Failed])
    C -.-> 1506
    1506 --> 1558
  
```

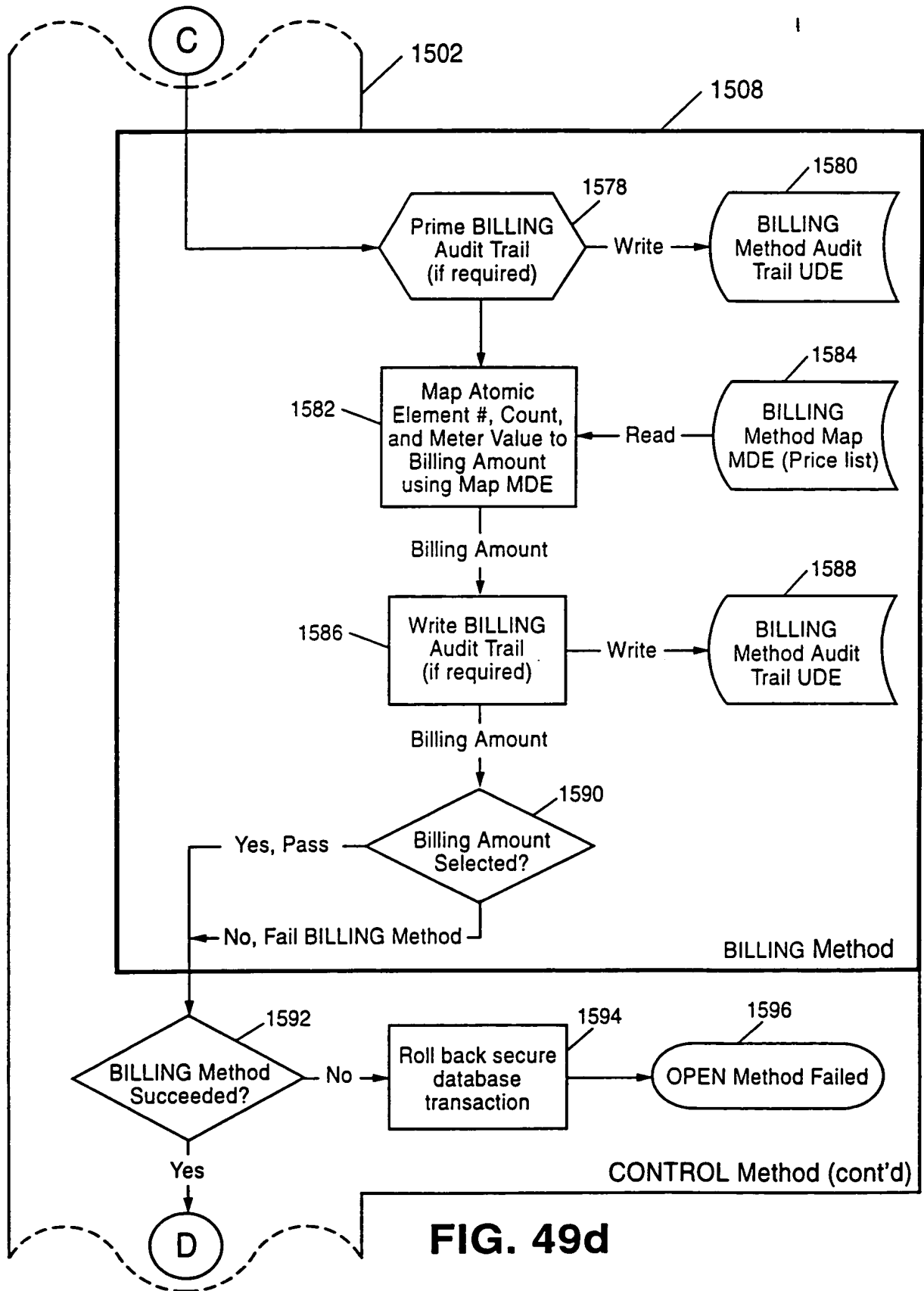
METER Method

CONTROL Method (cont'd)

FIG. 49c

FIG. 49c

09678252 "400300




```
graph TD
    D((D)) -- 1502 --> 1598{{Prime BUDGET Audit Trail (if required)}}
    1598 -- Write --> 1600[BUDGET Method Audit Trail UDE]
    1598 --> 1602[Add Billing Amount to Budget Value]
    1602 <--> |Read/Write| 1604[BUDGET Method UDE (the Budget)]
    1602 --> 1606[Write BUDGET Audit Trail (if required)]
    1606 -- Write --> 1608[BUDGET Method Audit Trail UDE]
    1606 --> 1610{BUDGET value out of range?}
    1610 -- "Yes, FAILS" --> 1612{BUDGET Method returns OK?}
    1610 -- "No, PASS" --> E((E))
    1612 -- No --> 1614[Roll back secure database transaction]
    1614 --> 1616([OPEN Method Failed])
    1612 -- Yes --> E
    E -- 1510 --> CONTROL[CONTROL Method (cont'd)]
```

FIG. 49e

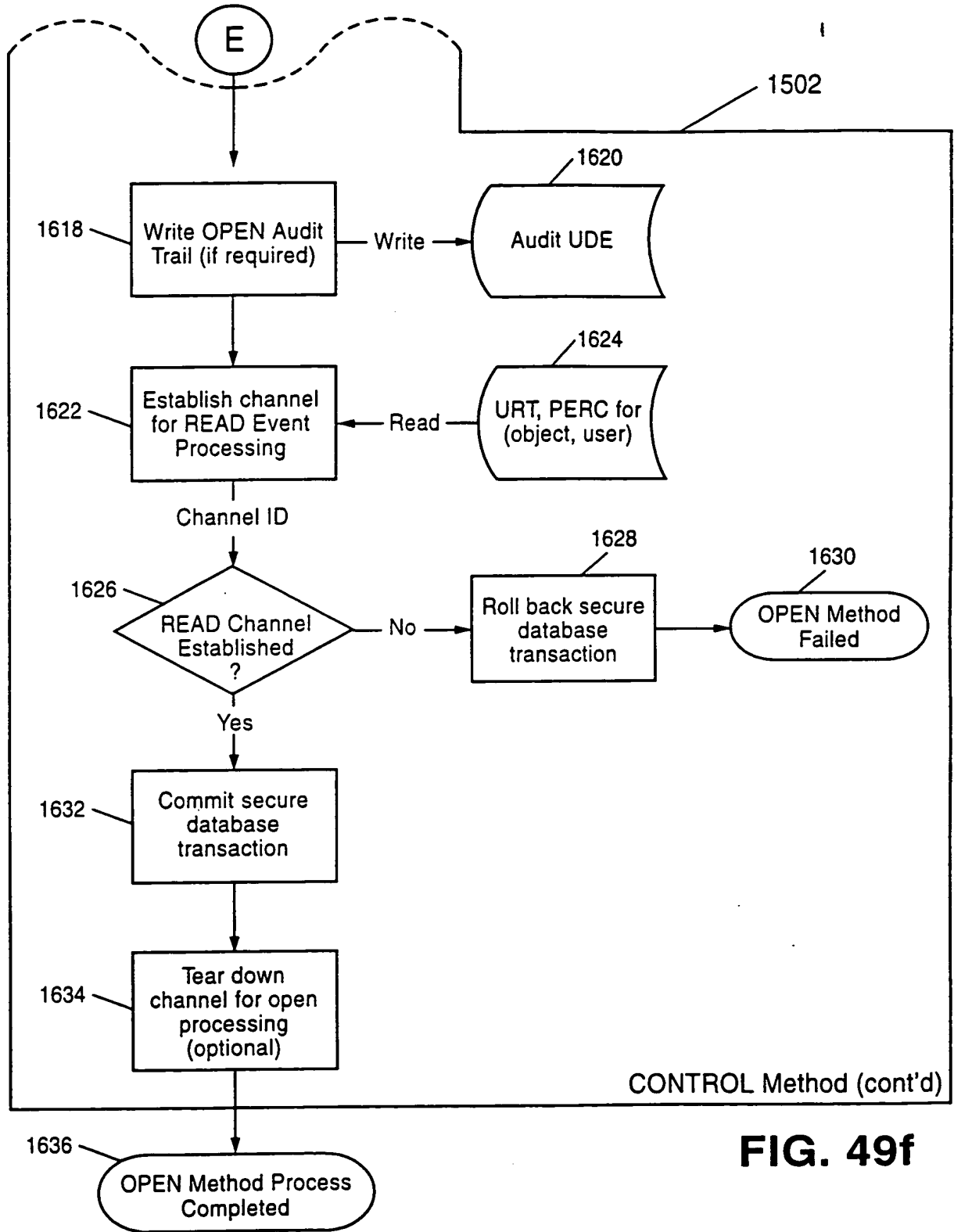


FIG. 49f

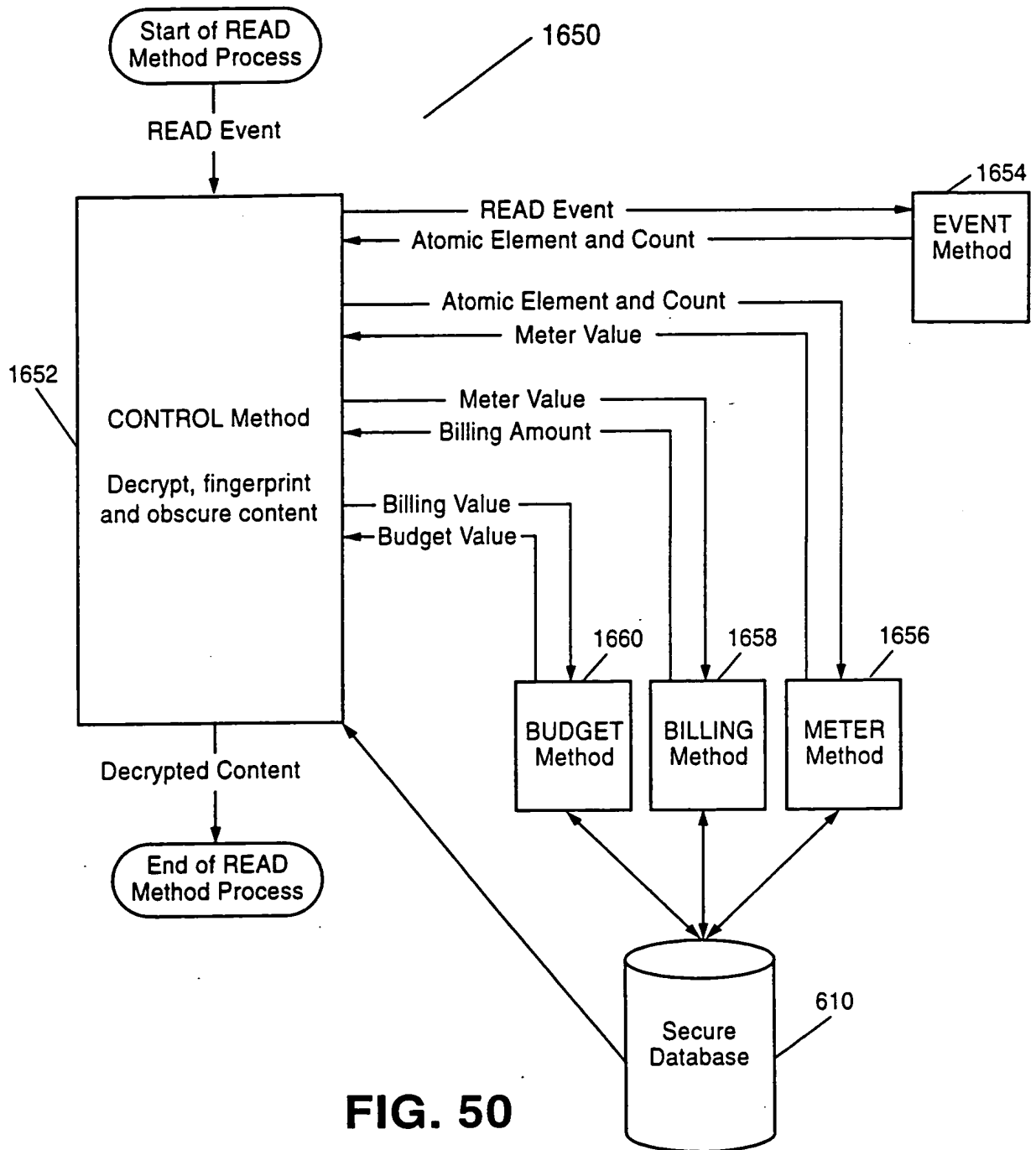


FIG. 50

1650

Start of READ
Method Process

1652

READ Event

Determine
identification of
object and user ID
for read

1662

READ Event, Object ID, User ID

Is
the Object
open for this
user?

1664

No

1666
Call the
OPEN
Method for
the Object.
Restart the
READ Method
once the
registration
is complete.

Yes

Prime Audit
(if required)

1670

Write

1672
Audit UDE

Start Secure
Database Transaction

1668

CONTROL Method

A

FIG. 50a

03070250 100300

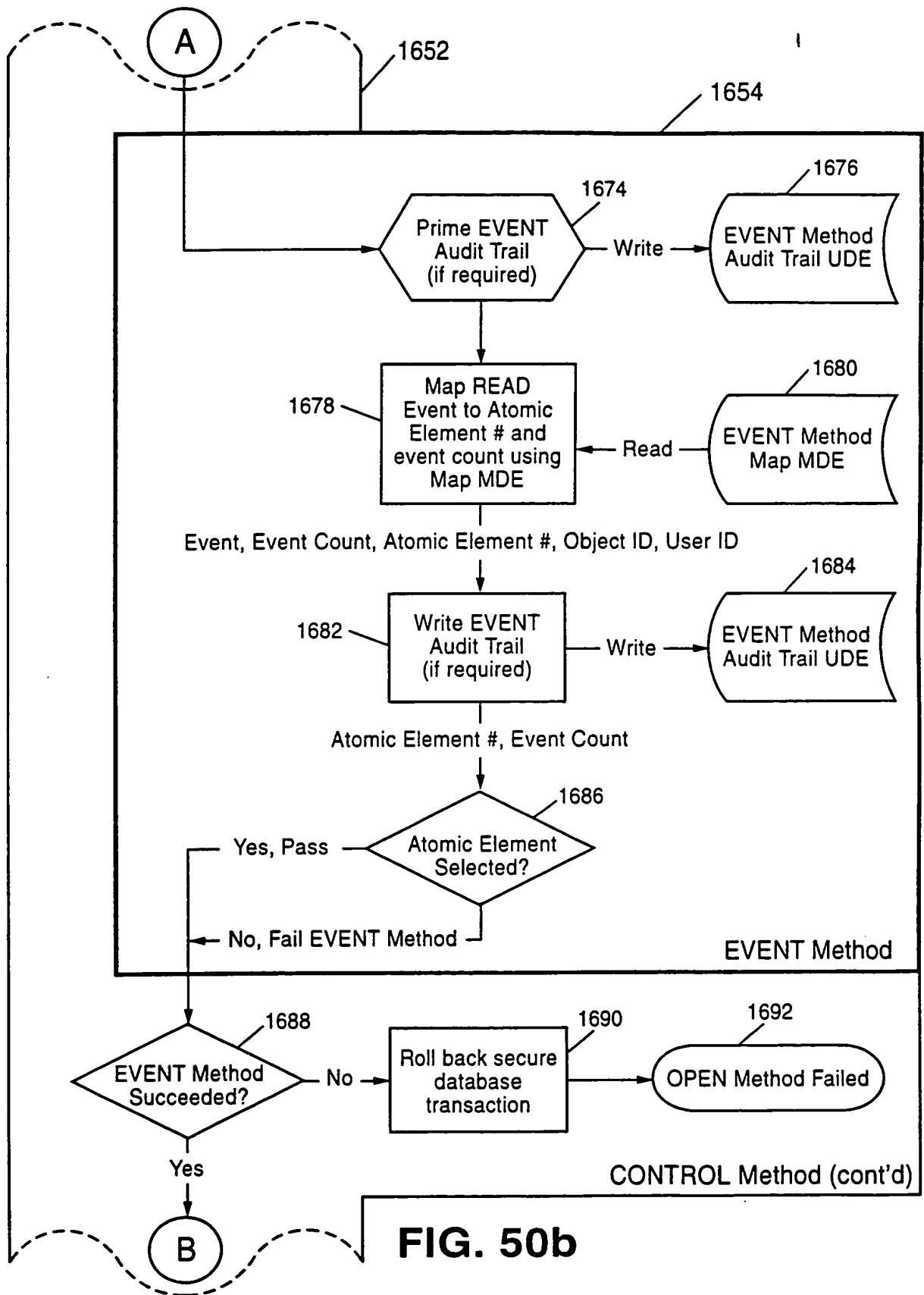


FIG. 50b

```

graph TD
    B((B)) --> 1694{Prime METER Audit Trail (if required)}
    1694 -- Write --> 1656[METER Method Audit Trail UDE]
    1694 --> 1698[Add EVENT Count to Meter Value]
    1698 <--> |Read/Write| 1700[METER Method UDE (the Meter)]
    1698 --> 1702[Write METER Audit Trail (if required)]
    1702 -- Write --> 1704[METER Method Audit Trail UDE]
    1702 -- METER Value --> 1706{Meter Increment Succeeded?}
    1706 -- "Yes, Pass" --> 1708{METER Method Succeeded?}
    1706 -- "No, Fail METER Method" --> B
    1708 -- Yes --> C((C))
    1708 -- No --> 1710[Roll back secure database transaction]
    1710 --> 1712([READ Method Failed])
    style B stroke-dasharray: 5 5
    style C stroke-dasharray: 5 5
    subgraph CONTROL_Method [CONTROL Method (cont'd)]
        1694
        1698
        1702
        1706
        1708
        1710
    end

```

FIG. 50c

FIG. 50c

```

graph TD
    C((C)) -- 1652 --> 1714{{Prime BILLING Audit Trail (if required)}}
    1714 -- Write --> 1716[BILLING Method Audit Trail UDE]
    1714 --> 1718[Map Atomic Element #, Count, and Meter Value to Billing Amount using Map MDE]
    1720[BILLING Method Map MDE (Price list)] -- Read --> 1718
    1718 -- Billing Amount --> 1722[Write BILLING Audit Trail (if required)]
    1722 -- Write --> 1724[BILLING Method Audit Trail UDE]
    1722 -- Billing Amount --> 1726{Billing Amount Selected?}
    1726 -- "Yes, Pass" --> 1728{BILLING Method Succeeded?}
    1726 -- "No, Fail BILLING Method" --> 1728
    1728 -- No --> 1730[Roll back secure database transaction]
    1730 --> 1732([READ Method Failed])
    1728 -- Yes --> D((D))
    D -- 1658 --> C
  
```

FIG. 50d

FIG. 50d

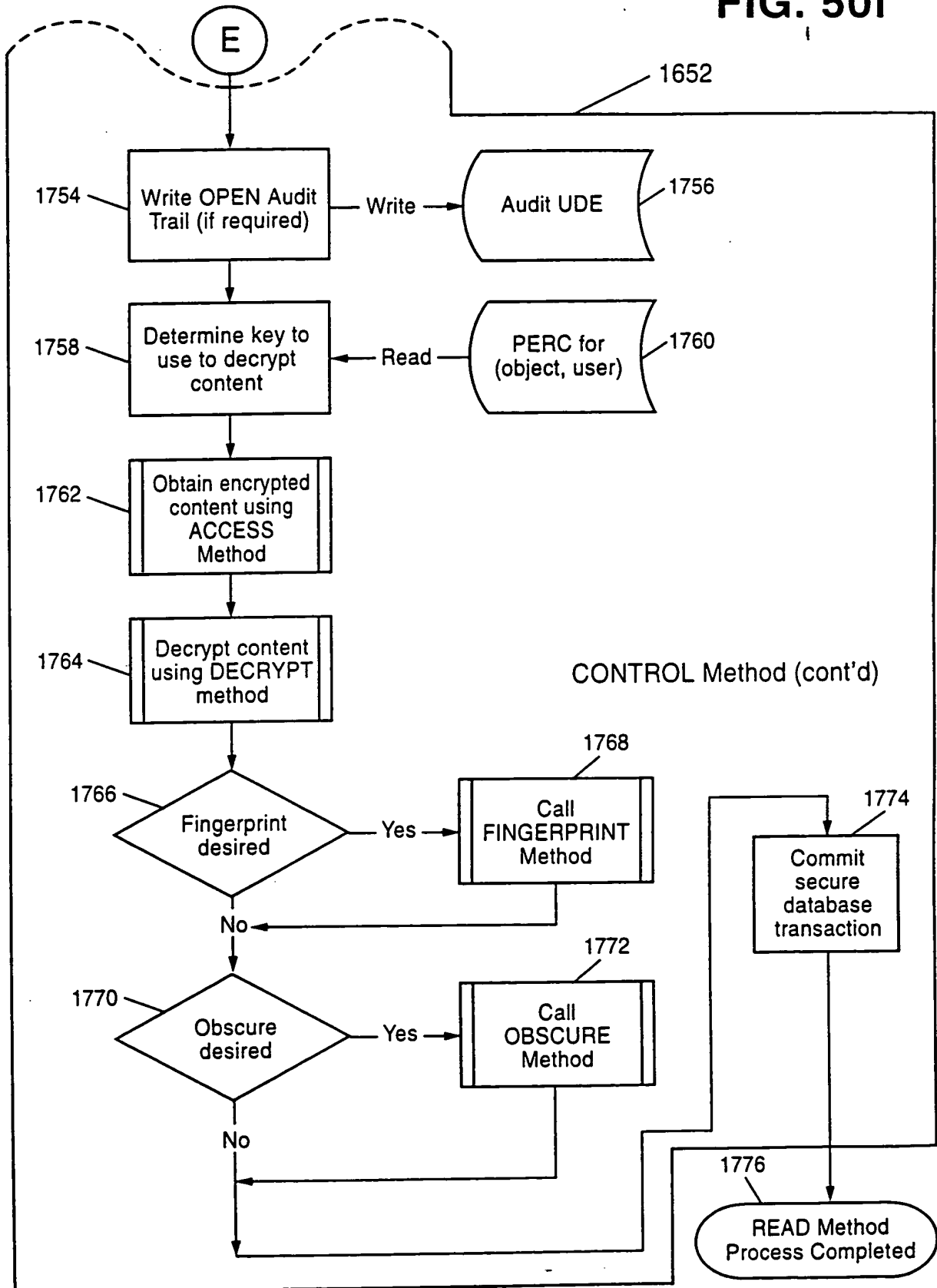
```

graph TD
    D((D)) -- 1562 --> 1734{{Prime BUDGET  
Audit Trail  
(if required)}}
    1734 -- Write --> 1736[BUDGET  
Method Audit  
Trail UDE]
    1734 --> 1738[Add Billing  
Amount to  
Budget Value]
    1738 <--> |Read/Write| 1740[BUDGET  
Method UDE  
(the Budget)]
    1738 --> 1742[Write BUDGET  
Audit Trail  
(if required)]
    1742 -- Write --> 1744[BUDGET  
Method Audit  
Trail UDE]
    1742 --> 1746{BUDGET  
value out of  
range?}
    1746 -- "Yes, FAILS" --> 1748{BUDGET  
Method returns  
OK?}
    1746 -- "No, PASS" --> 1748
    1748 -- No --> 1706[Roll back secure  
database  
transaction]
    1706 --> 1752([READ Method Failed])
    1748 -- Yes --> E((E))
    E -- 1660 --> D
    subgraph BUDGET_Method [BUDGET Method]
        1734
        1738
        1742
        1746
    end
    subgraph CONTROL_Method [CONTROL Method (cont'd)]
        1706
        1752
    end

```

FIG. 50e

FIG. 50f



2025 RELEASE UNDER E.O. 14176

```

graph TD
    Start([Start of WRITE Method Process]) --> WriteEvent[WRITE Event]
    WriteEvent --> ControlMethod[CONTROL Method]
    ControlMethod --> EventMethod[EVENT Method]
    EventMethod --> ControlMethod
    EventMethod --> ControlMethod
    ControlMethod --> BudgetMethod[BUDGET Method]
    ControlMethod --> BillingMethod[BILLING Method]
    ControlMethod --> MeterMethod[METER Method]
    ControlMethod --> End([End of WRITE Method Process])
    BudgetMethod <--> SecureDatabase[(Secure Database)]
    BillingMethod <--> SecureDatabase
    MeterMethod <--> SecureDatabase
    ControlMethod --> SecureDatabase

```

FIG. 51

FIG. 51

```
graph TD
    1792([Start of WRITE Method Process]) --> 1782[WRITE Event]
    1782 --> 1794[Determine identification of object and user ID for read]
    1794 --> 1795[WRITE Event, Object ID, User ID]
    1795 --> 1796{Is the Object open for this user?}
    1796 -- No --> 1798[Call the OPEN Method for the Object. Restart the WRITE Method once the registration is complete.]
    1798 --> 1782
    1796 -- Yes --> 1802{{Prime Audit (if required)}}
    1802 -- Write --> 1804[(Audit UDE)]
    1802 --> 1800[Start Secure Database Transaction]
    1800 --> A((A))
    A -.-> 1782
```

FIG. 51a

```

graph TD
    A((A)) -- 1782 --> 1806{{Prime EVENT Audit Trail (if required)}}
    1806 -- Write --> 1808[(EVENT Method Audit Trail UDE)]
    1806 --> 1810[Map WRITE Event to Atomic Element # and event count using Map MDE]
    1812[(EVENT Method Map MDE)] -- Read --> 1810
    1810 -- "Event, Event Count, Atomic Element #, Object ID, User ID" --> 1814[Write EVENT Audit Trail (if required)]
    1814 -- Write --> 1816[(EVENT Method Audit Trail UDE)]
    1814 -- "Atomic Element #, Event Count" --> 1818{Atomic Element Selected?}
    1818 -- "Yes, Pass" --> 1822{EVENT Method Succeeded?}
    1818 -- No --> 1820[Update EVENT Method Map MDE to reflect new data]
    1820 -- "PASS if update succeeded, FAIL otherwise" --> 1822
    1822 -- Yes --> B((B))
    1822 -- No --> 1824[Roll back secure database transaction]
    1824 --> 1826([WRITE Method Failed])
    B -.-> A
    
```

CONTROL Method (cont'd)

FIG. 51b

FIG. 51b

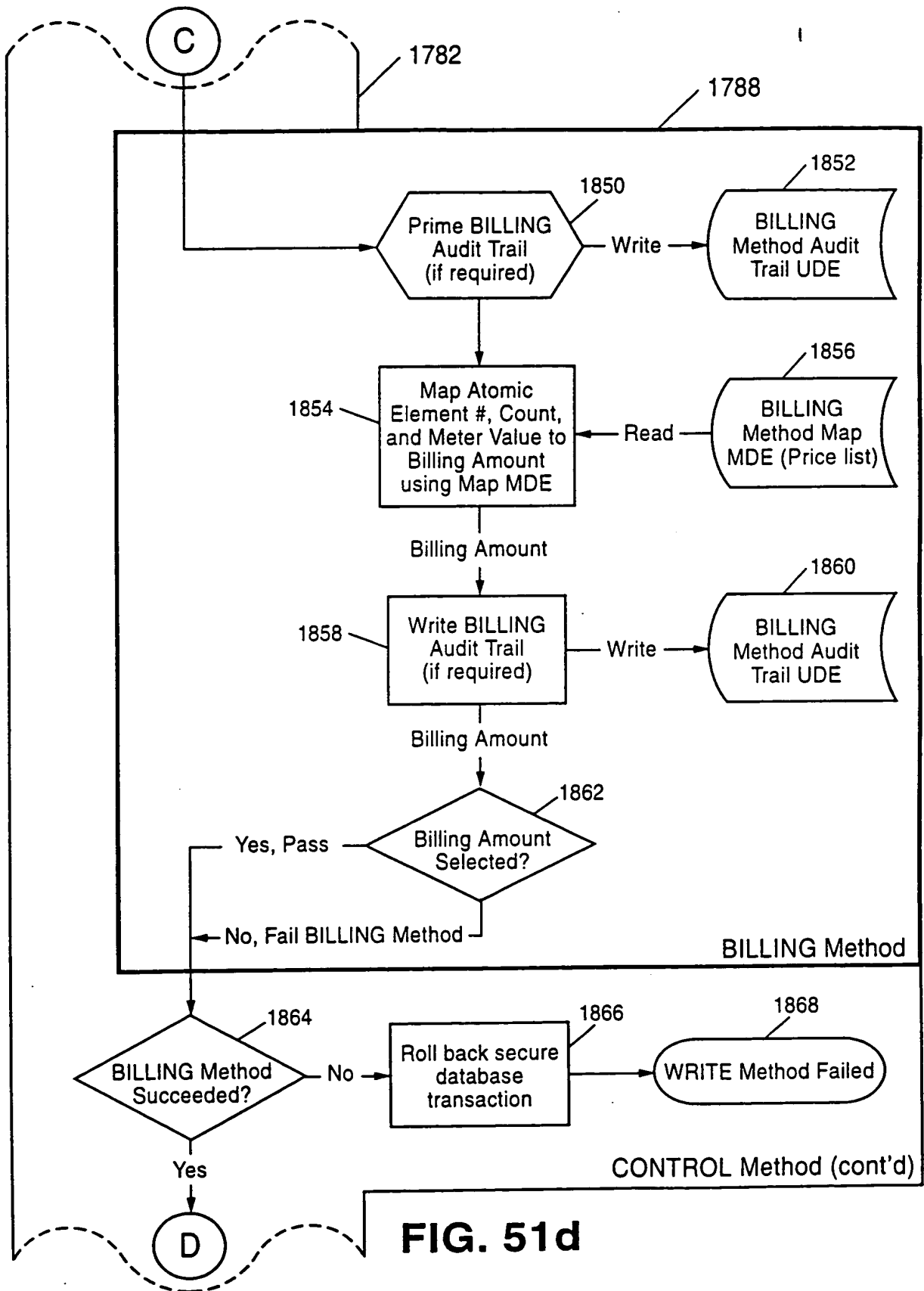
```

graph TD
    B((B)) -- 1782 --> 1828{{Prime METER Audit Trail  
(if required)}}
    1828 -- Write --> 1830[METER Method Audit Trail UDE]
    1828 --> 1834[Add EVENT Count to  
Meter Value]
    1834 <--> |Read/Write| 1836[METER Method UDE  
(the Meter)]
    1834 --> 1838[Write METER Audit Trail  
(if required)]
    1838 -- Write --> 1840[METER Method Audit Trail UDE]
    1838 -- METER Value --> 1842{Meter Increment  
Succeeded ?}
    1842 -- "Yes, Pass" --> 1844{METER Method Succeeded?}
    1842 -- "No, Fail METER Method" --> 1844
    1844 -- No --> 1846[Roll back secure database transaction]
    1846 --> 1848([WRITE Method Failed])
    1844 -- Yes --> C((C))
    C -- 1786 --> 1828
    subgraph CONTROL_Method [CONTROL Method (cont'd)]
        C
    end
  
```

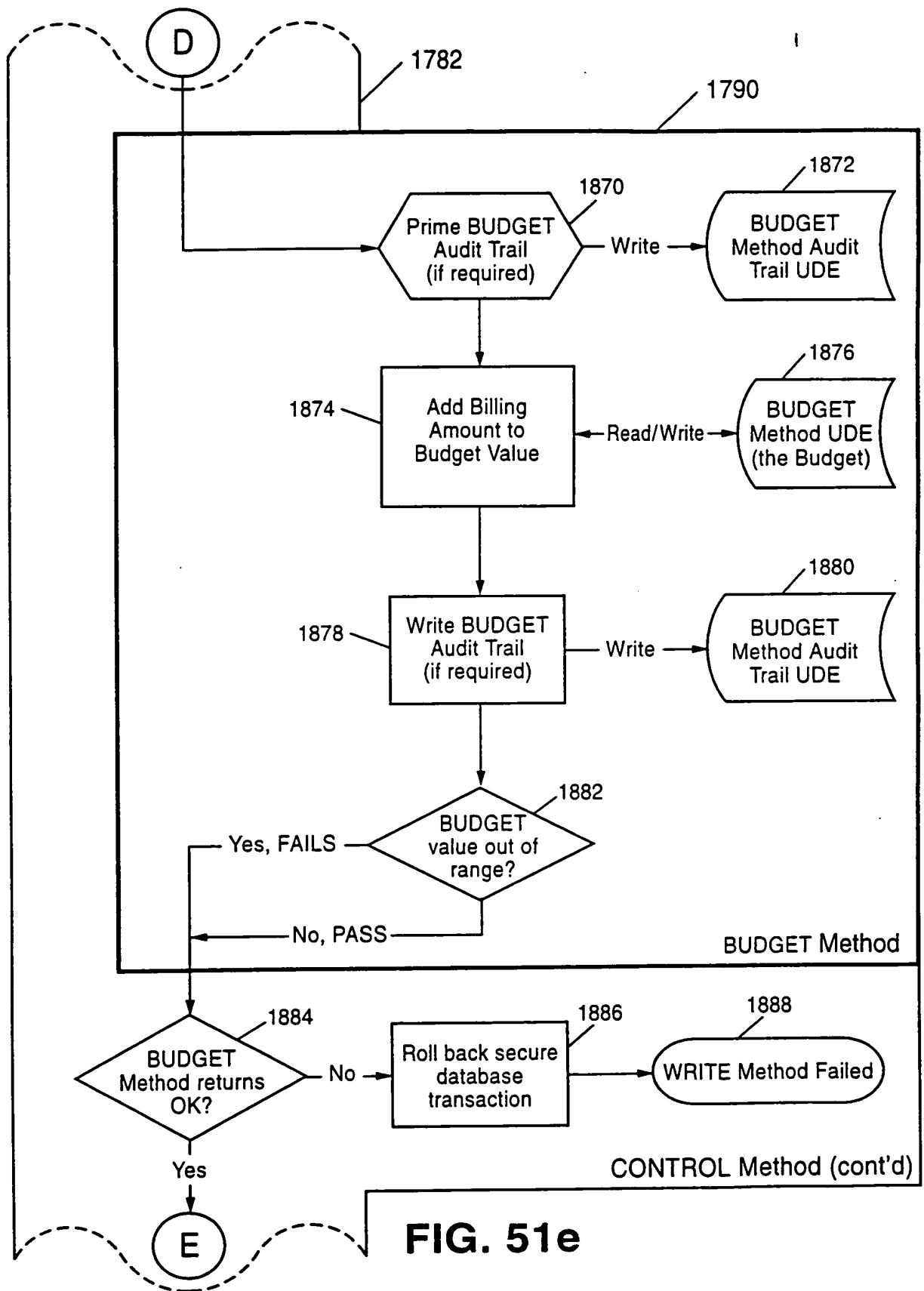
FIG. 51c

FIG. 51c

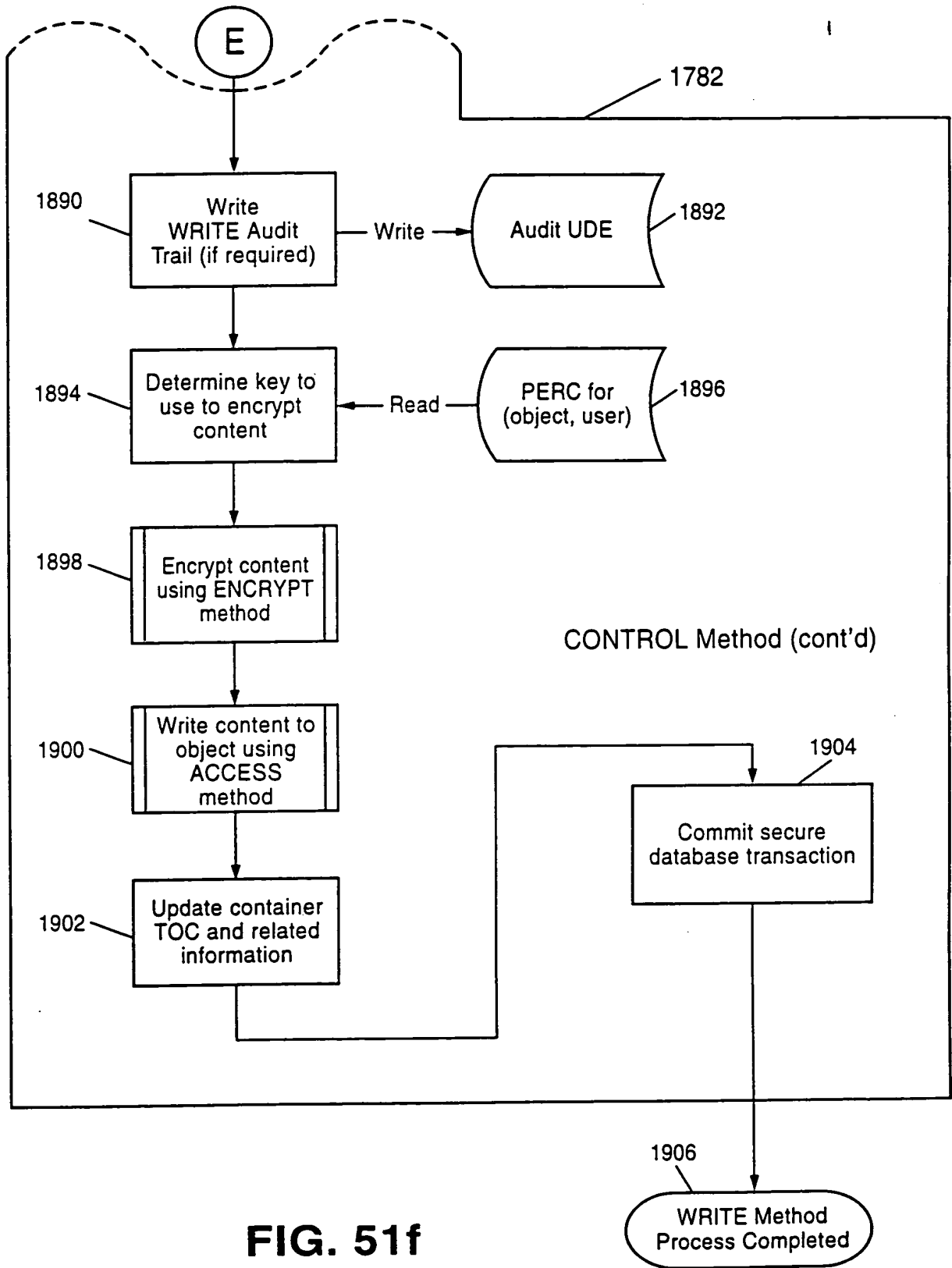
09676252 100300



09676252 100340



00000000 1003000




```

graph TD
    Start([EVENT Method Start]) -->|EVENT, Event Count, Event Parameters| 1942{{Prime EVENT Audit Trail (if required)}}
    1942 -- Write --> 1944[(EVENT Method Audit Trail UDE)]
    1942 --> 1946[Load MAP MDE DTD]
    1948[(EVENT Method Map DTD)] -- Read --> 1946
    1946 --> 1950[Map Event to Atomic Element # and event count using Map MDE]
    1952[(EVENT Method Map MDE)] -- Read --> 1950
    1950 -- "Event, Event Count, Atomic Element #, Object ID, User ID" --> 1970[Write EVENT Audit Trail (if required)]
    1970 -- Write --> 1972[(EVENT Method Audit Trail UDE)]
    1970 -- "Atomic Element #, Event Count" --> 1974{Atomic Element Selected?}
    1974 -- No --> 1976([EVENT Method failed])
    1974 --> 1978([EVENT Method Succeeded])

```

FIG. 53a

FIG. 53a

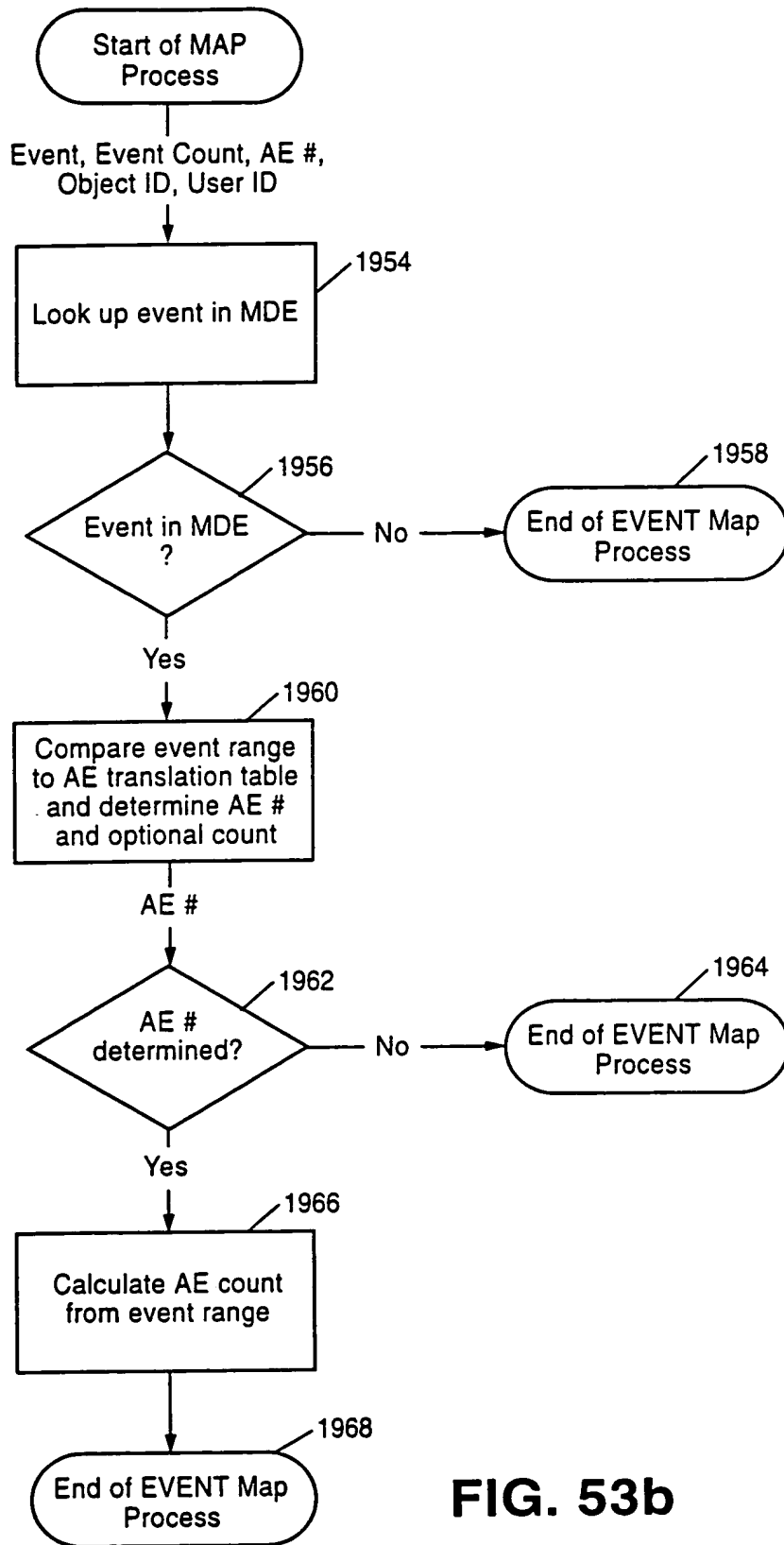


FIG. 53b

```

graph TD
    Start([BILLING Method Start]) -- Meter Value --> 1982{{Prime BILLING Audit Trail (if required)}}
    1982 -- Write --> 1984[BILLING Method Audit Trail UDE]
    1982 --> 1985[Load MAP MDE DTD]
    1986[BILLING Method Map DTD] -- Read --> 1985
    1985 --> 1988[Map meter value to billing amount using Map MDE (and possibly database elements)]
    1989[BILLING Method Map MDE (and optionally others)] -- Read --> 1988
    1988 -- Billing Amount --> 1990[Write BILLING Audit Trail (if required)]
    1990 -- Write --> 1992[BILLING Method Audit Trail UDE]
    1990 --> 1994{Billing Amount Selected?}
    1994 -- No --> 1996([BILLING Method failed])
    1994 -- Billing Amount --> 1998([BILLING Method Succeeded])
  
```

FIG. 53c

FIG. 53c

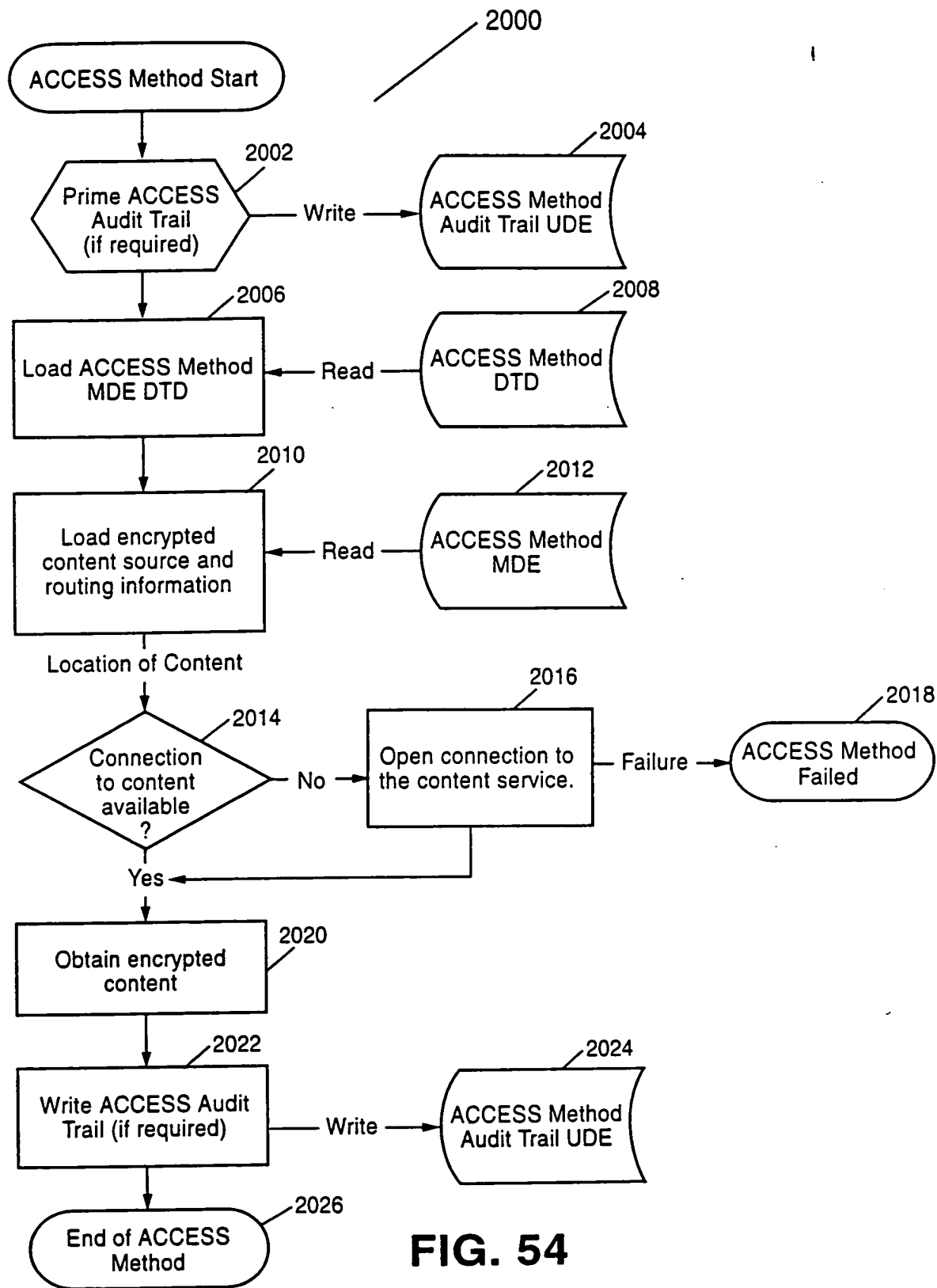


FIG. 54

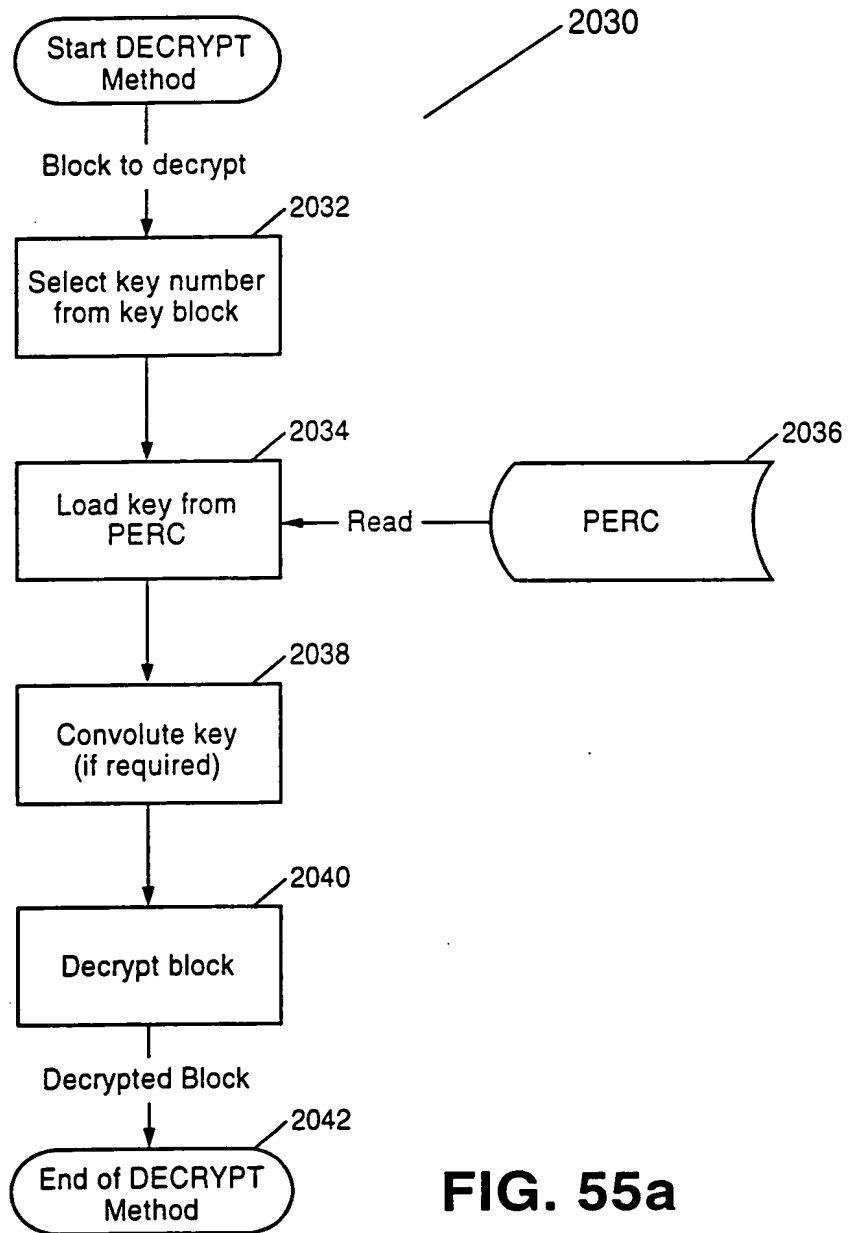


FIG. 55a

```
graph TD; 2050([Start ENCRYPT Method]) --> 2052[Block to Encrypt]; 2052 --> 2054[Determine key to use from key block]; 2054 --> 2056[Load key from PERC or Add key to PERC]; 2056 <--> |Read/Write| 2056a[(PERC)]; 2056 --> 2058[Convolute key (if required)]; 2058 --> 2060[Encrypt block]; 2060 --> 2062[Encrypted Block]; 2062 --> 2064([End of ENCRYPT block]);
```

FIG. 55b

FIG. 55b

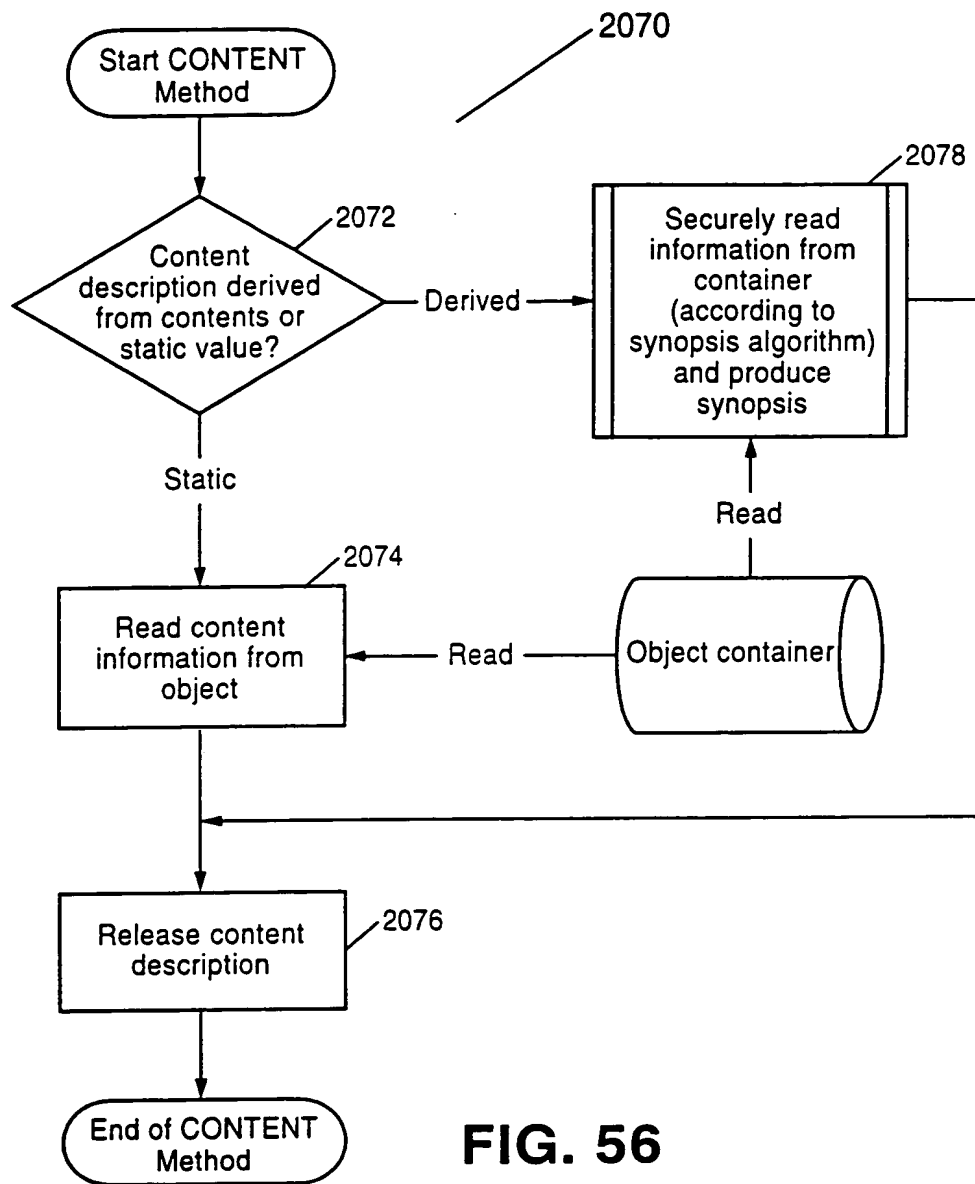


FIG. 56

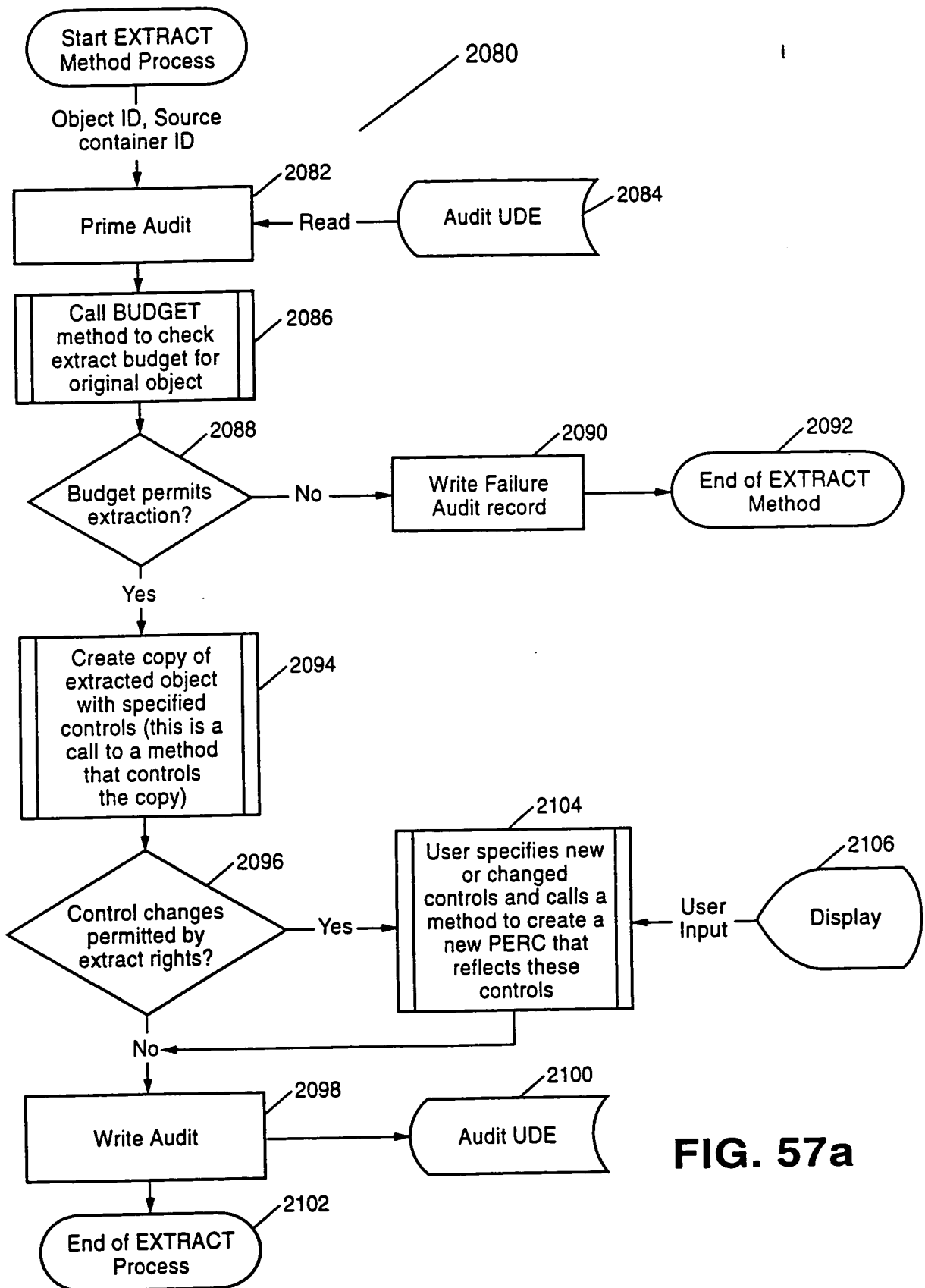
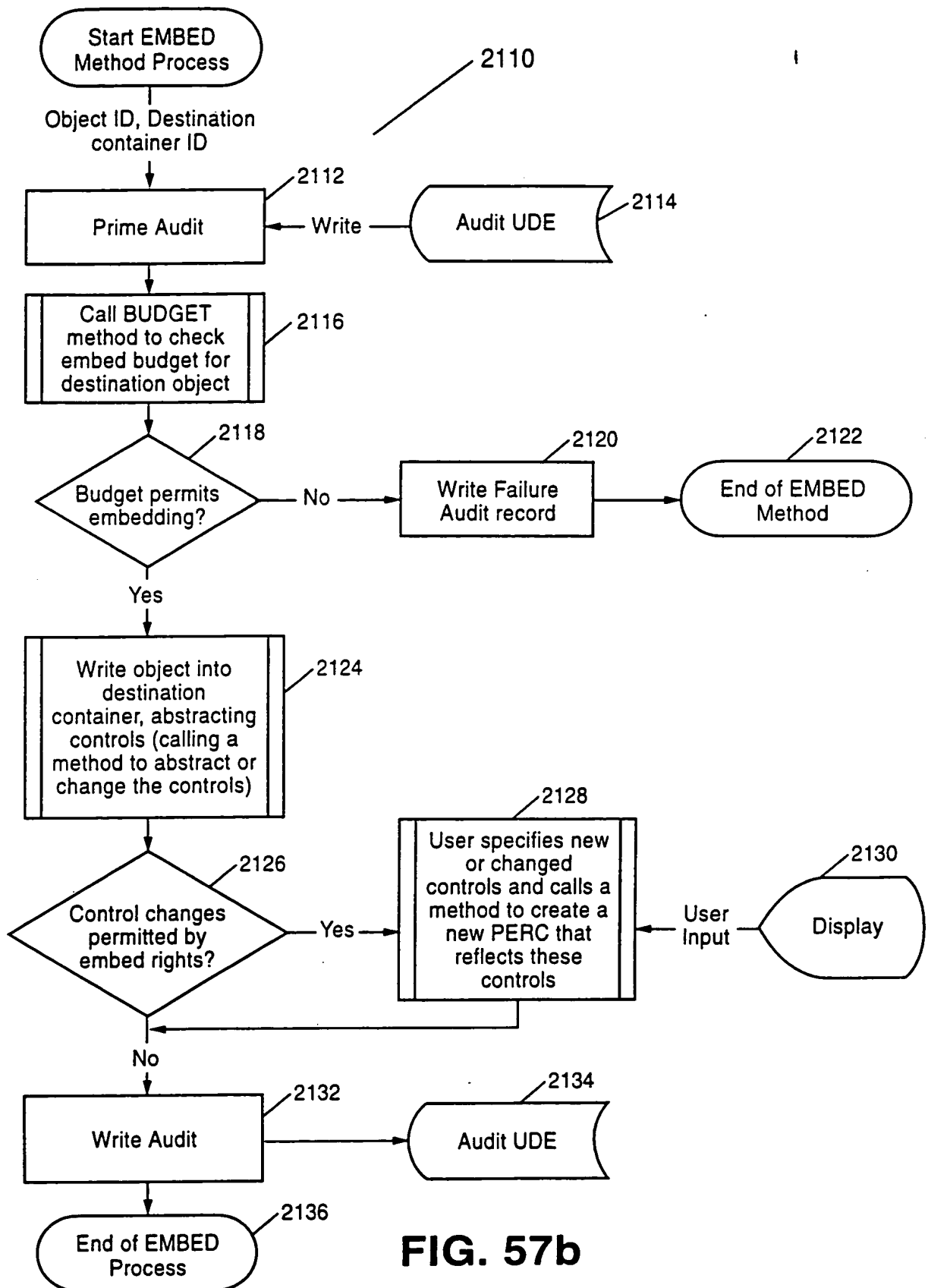


FIG. 57a



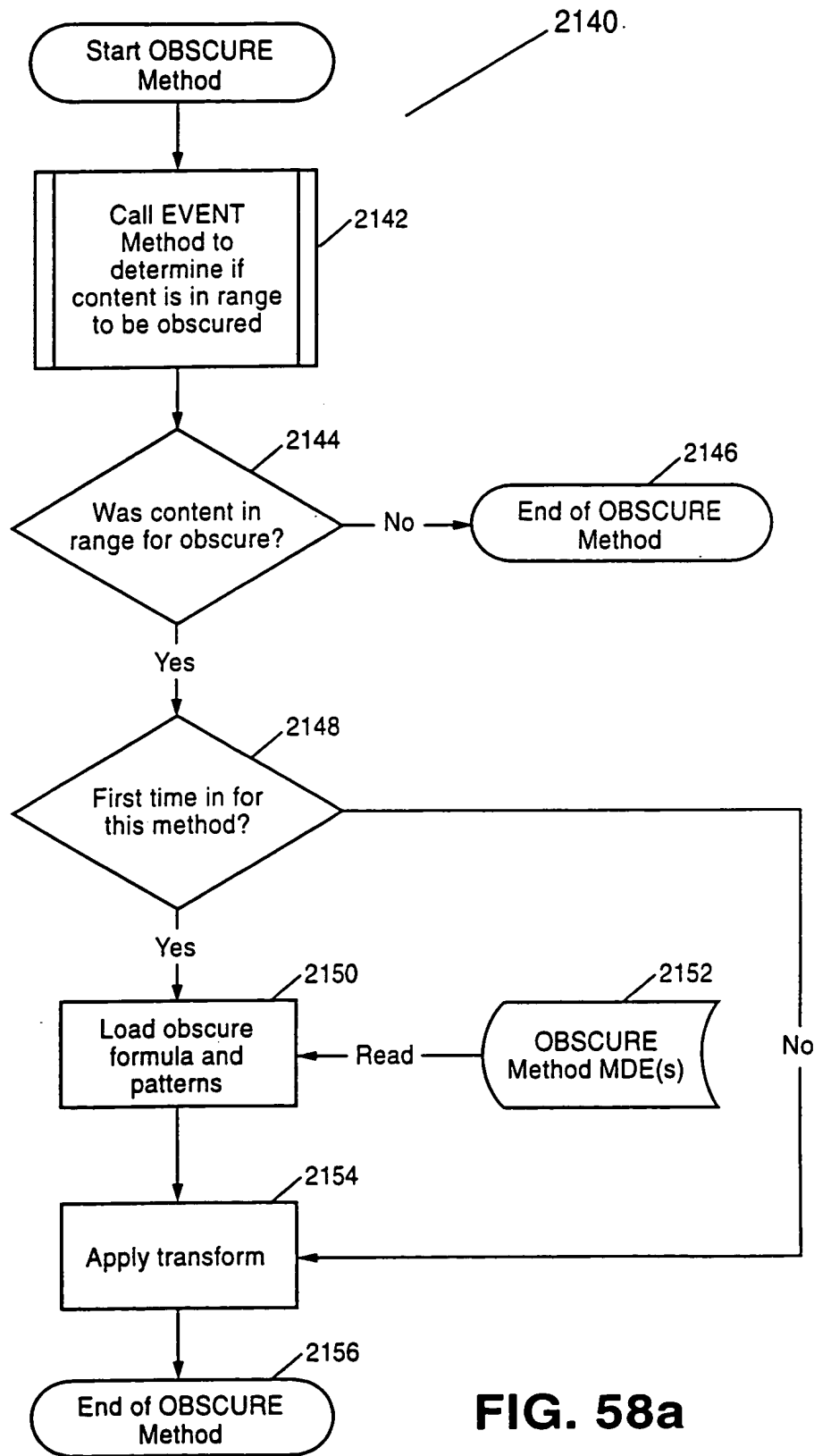


FIG. 58a

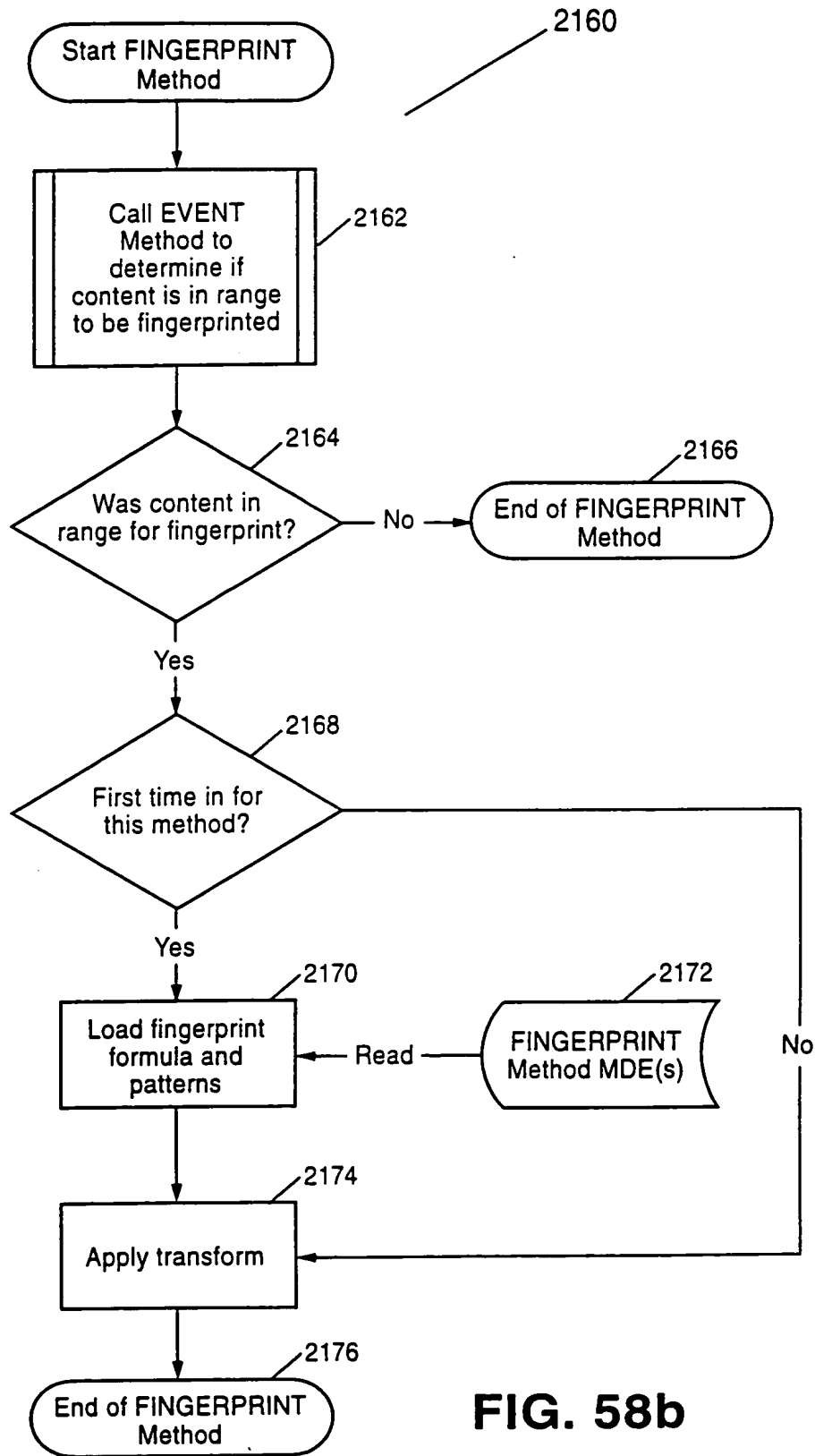


FIG. 58b

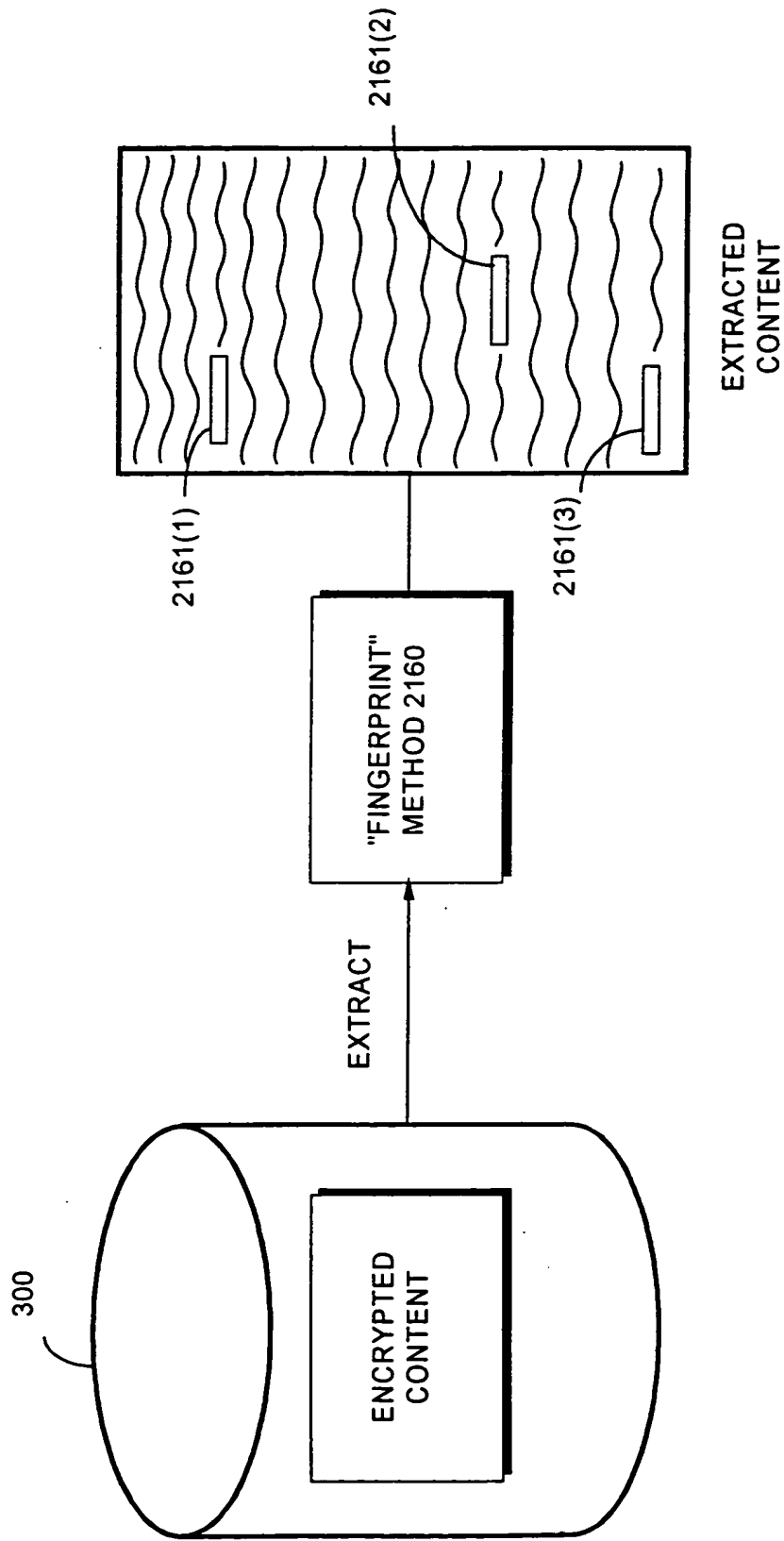


FIG. 58C

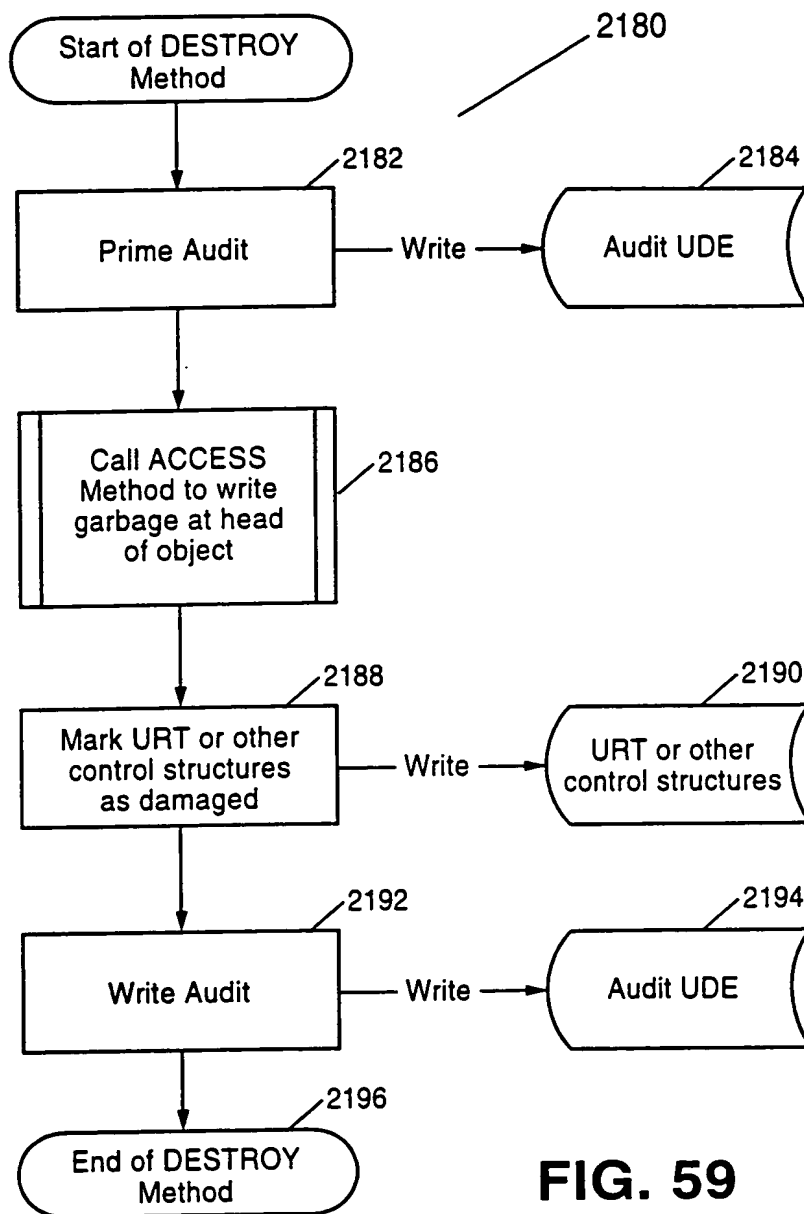


FIG. 59

09676252 100399

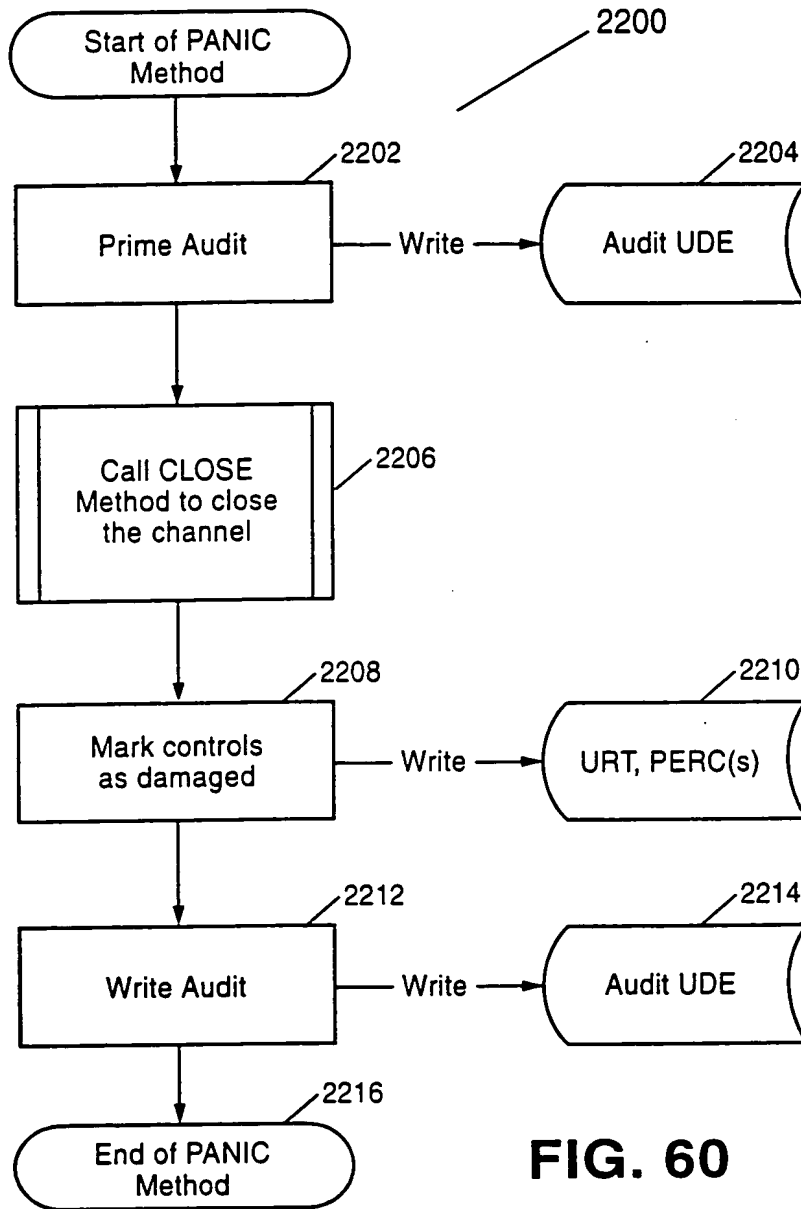


FIG. 60

```

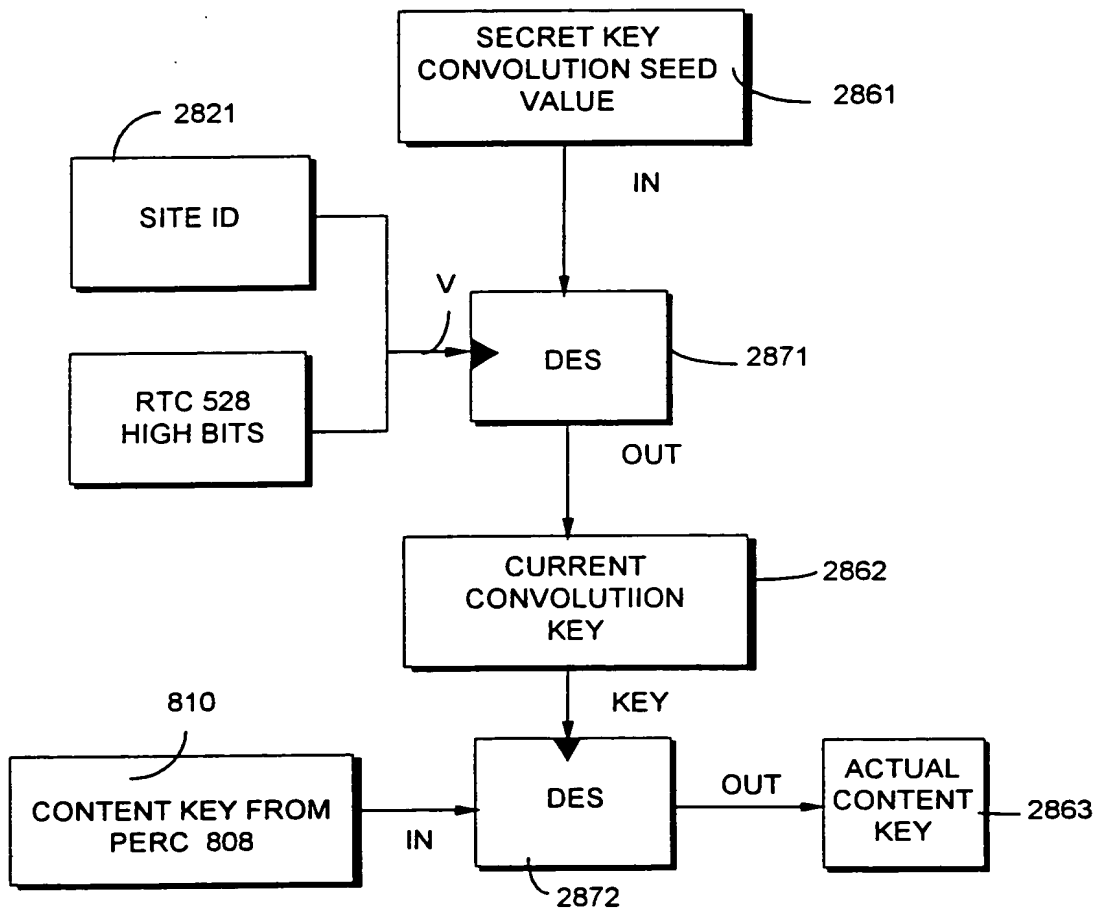
graph TD
    Start([Start METER Method Use Process]) --> Input[Atomic Element, Event Count]
    Input --> 2222[Prime METER Audit Trail]
    2222 -- Write --> 2224[METER Audit Trail UDE]
    2222 --> 2226[Obtain DTD for METER]
    2228[DTD for METER UDE] -- Read --> 2226
    2226 --> 2230[Obtain METER]
    2232[METER UDE] -- Read --> 2230
    2230 --> 2234{METER Audit date expired (time)?}
    2234 -- Yes --> 2236[Commit METER Failure Audit Record]
    2236 --> 2238([METER Method Failed])
    2234 -- No --> 2239[Update METER using Atomic Element and count]
    2239 -- Write --> 2240[METER UDE]
    2239 --> 2242[Save METER Use Audit Record]
    2242 -- Write --> 2244[METER Audit Trail UDE]
    2242 --> 2246([METER Method Succeeded])

```

FIG. 61

FIG. 61

FIG. 62



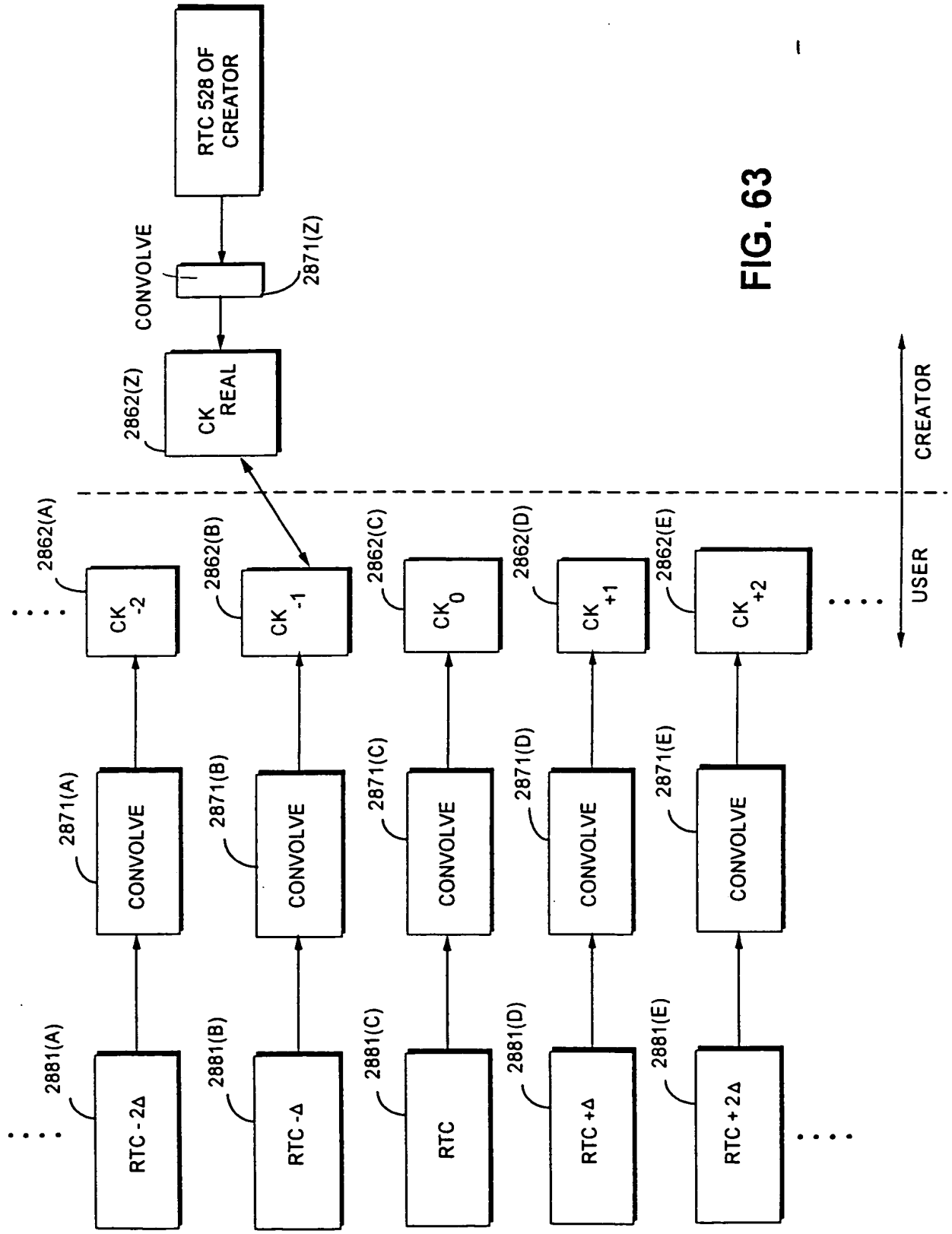
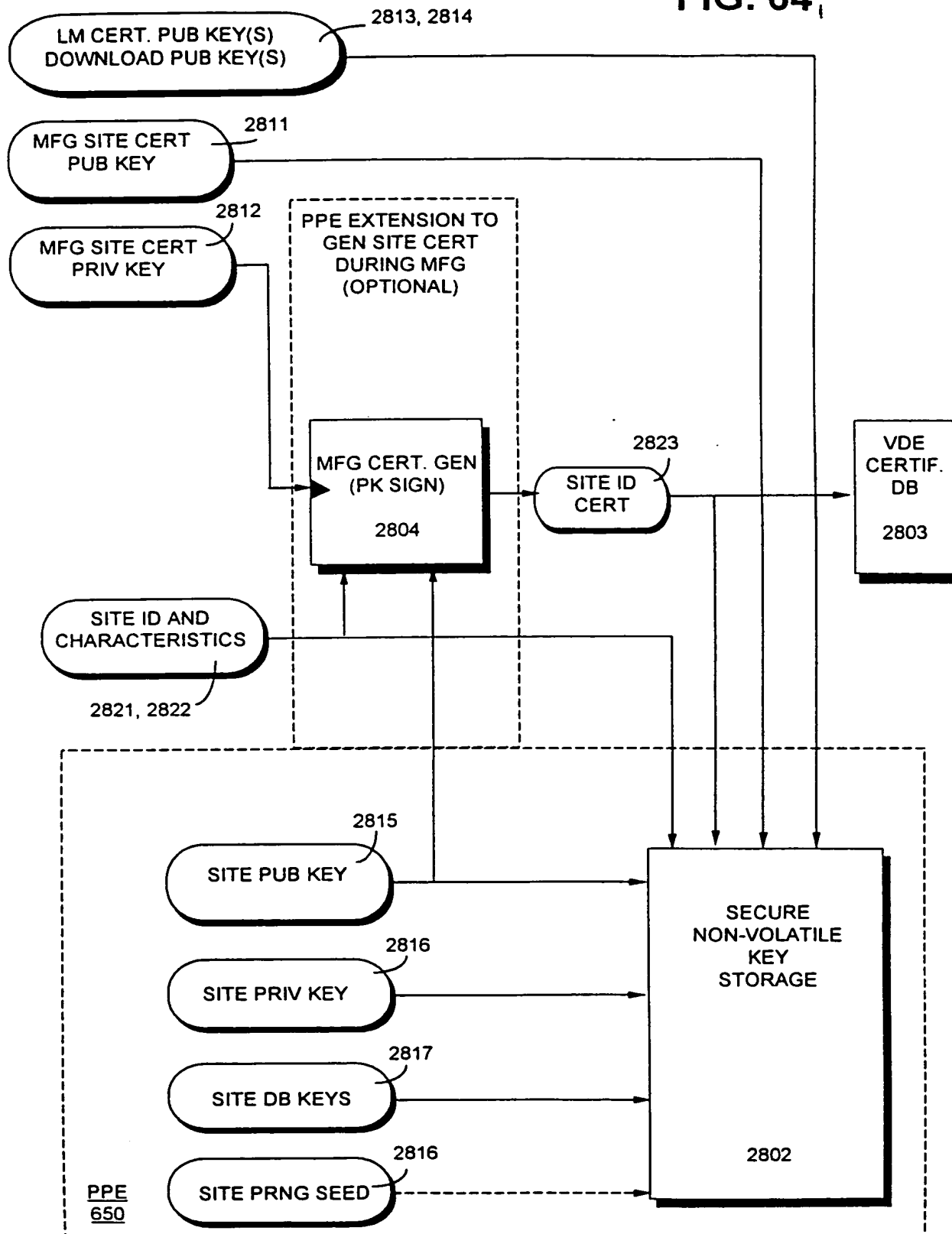


FIG. 63

FIG. 64



2025 RELEASE UNDER E.O. 14176

FIG. 65

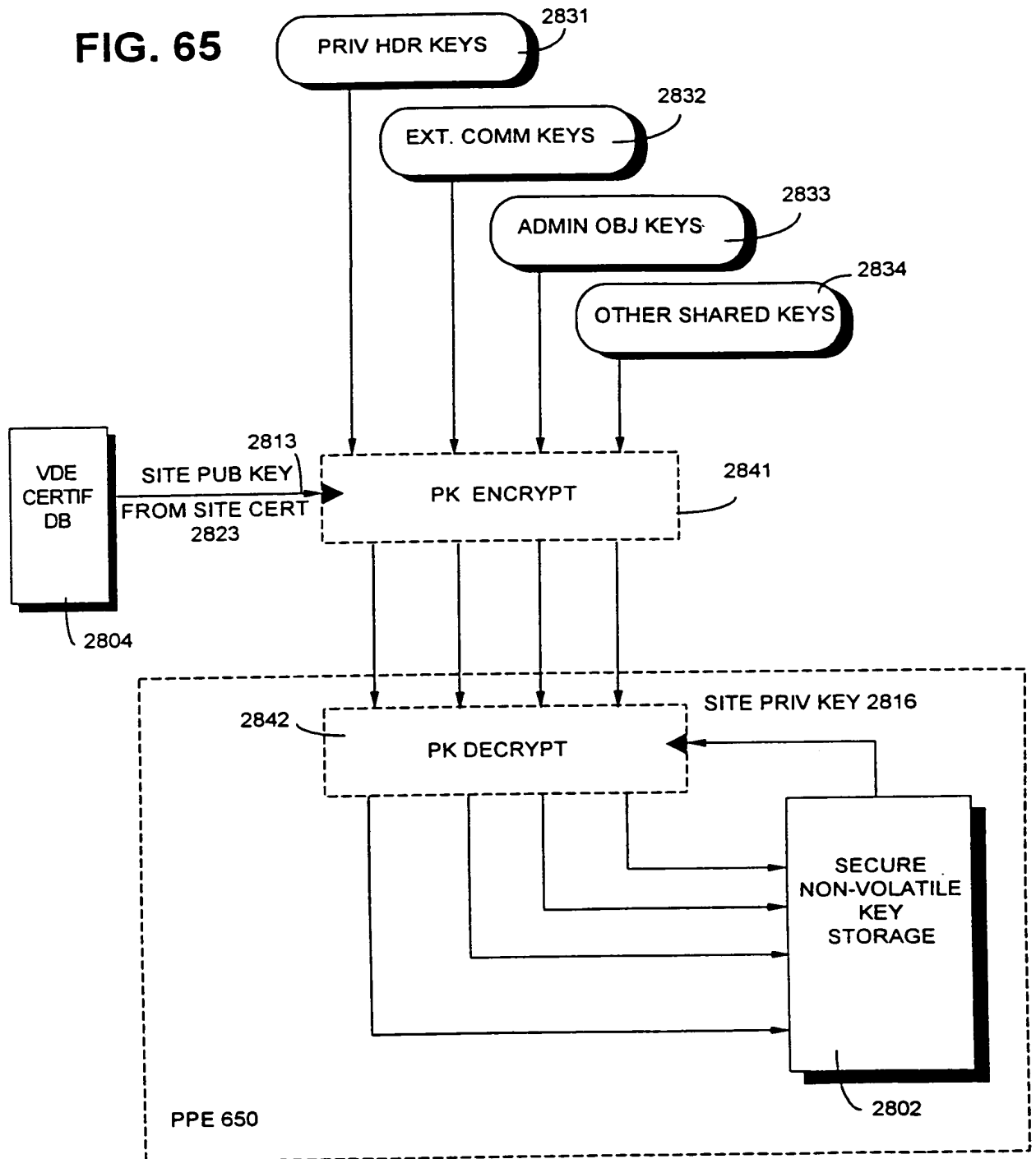
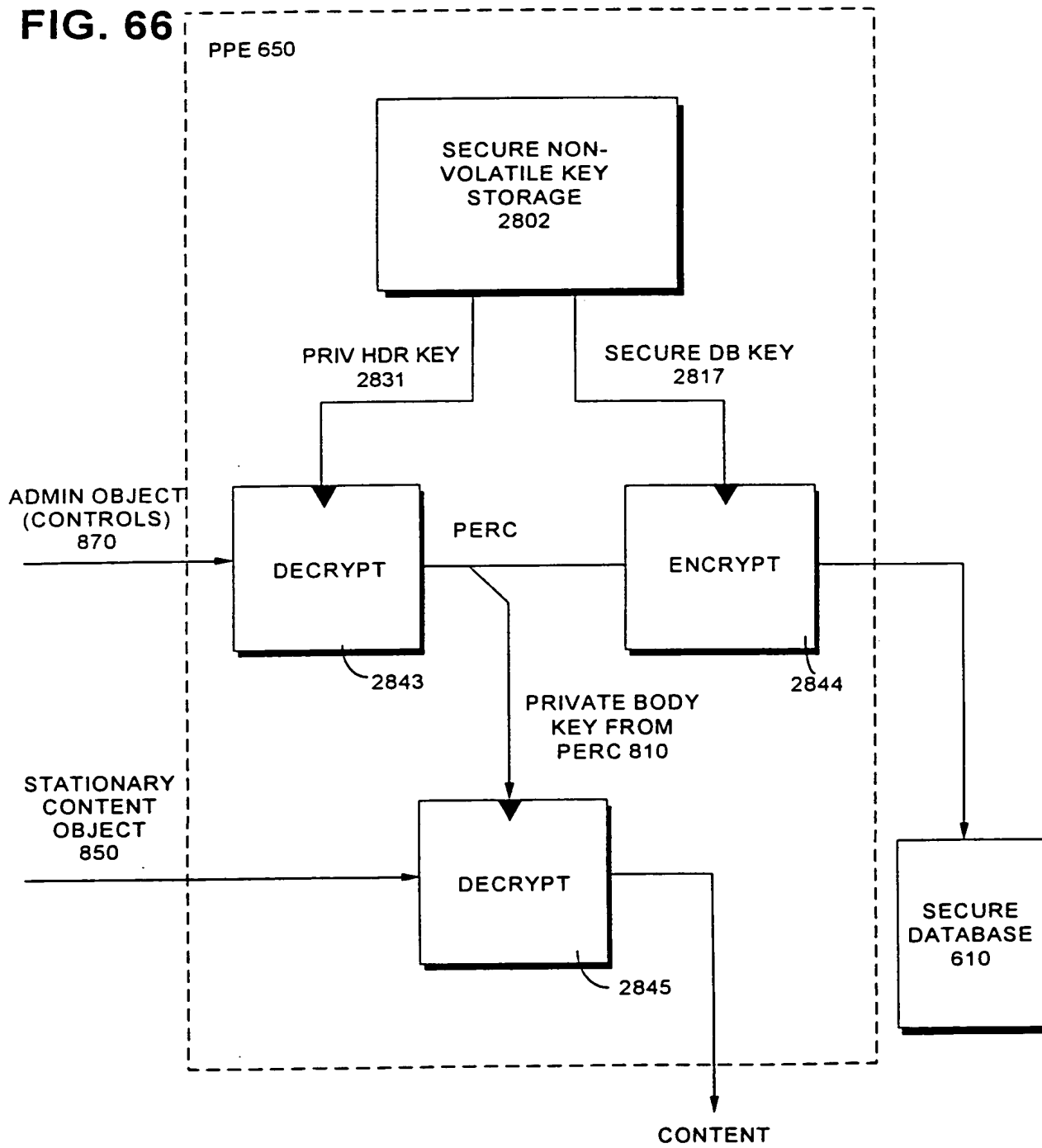


FIG. 66



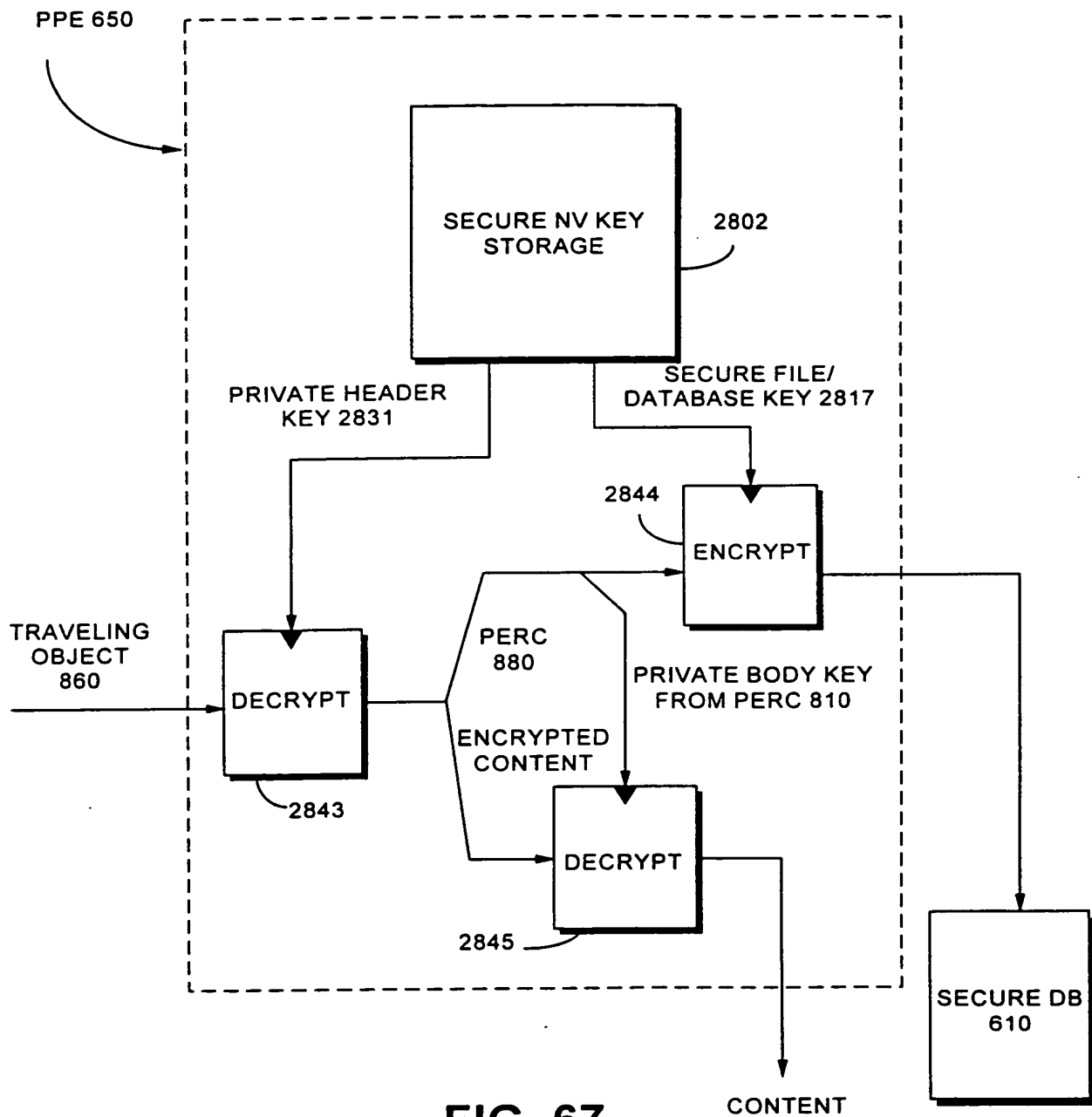


FIG. 67

FIG. 68

1370

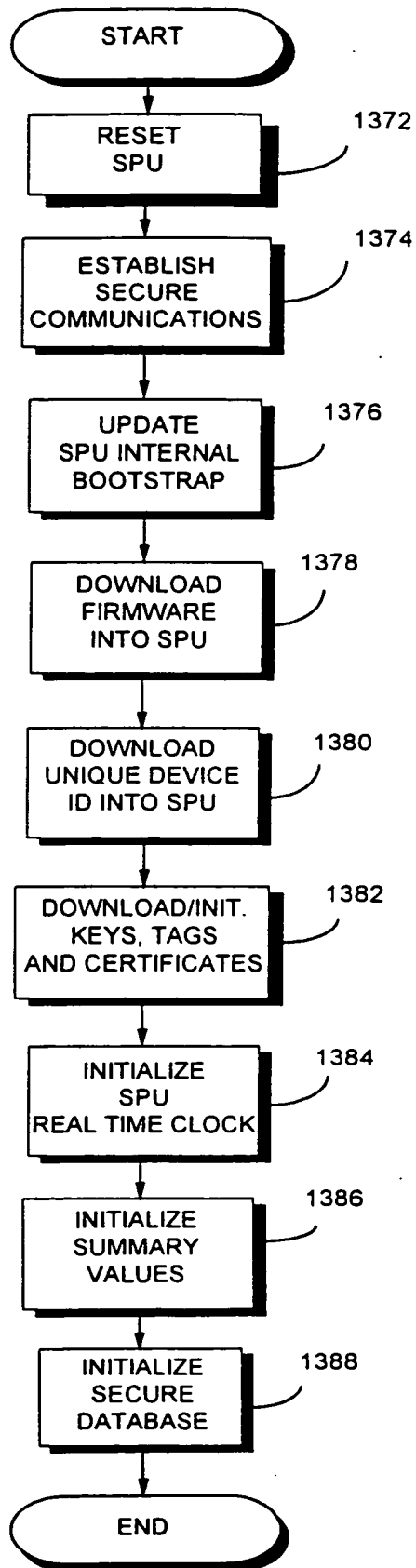
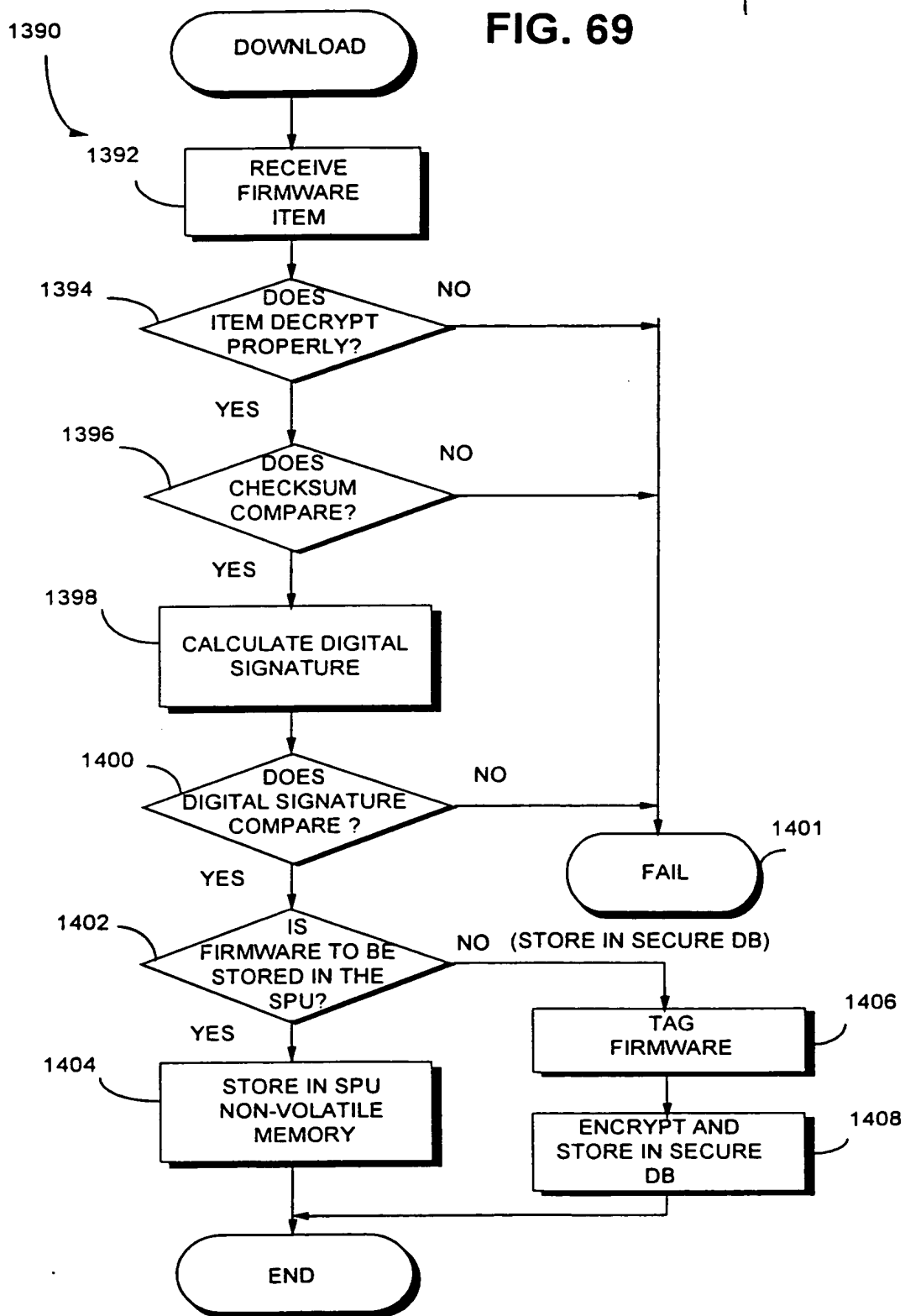


FIG. 69



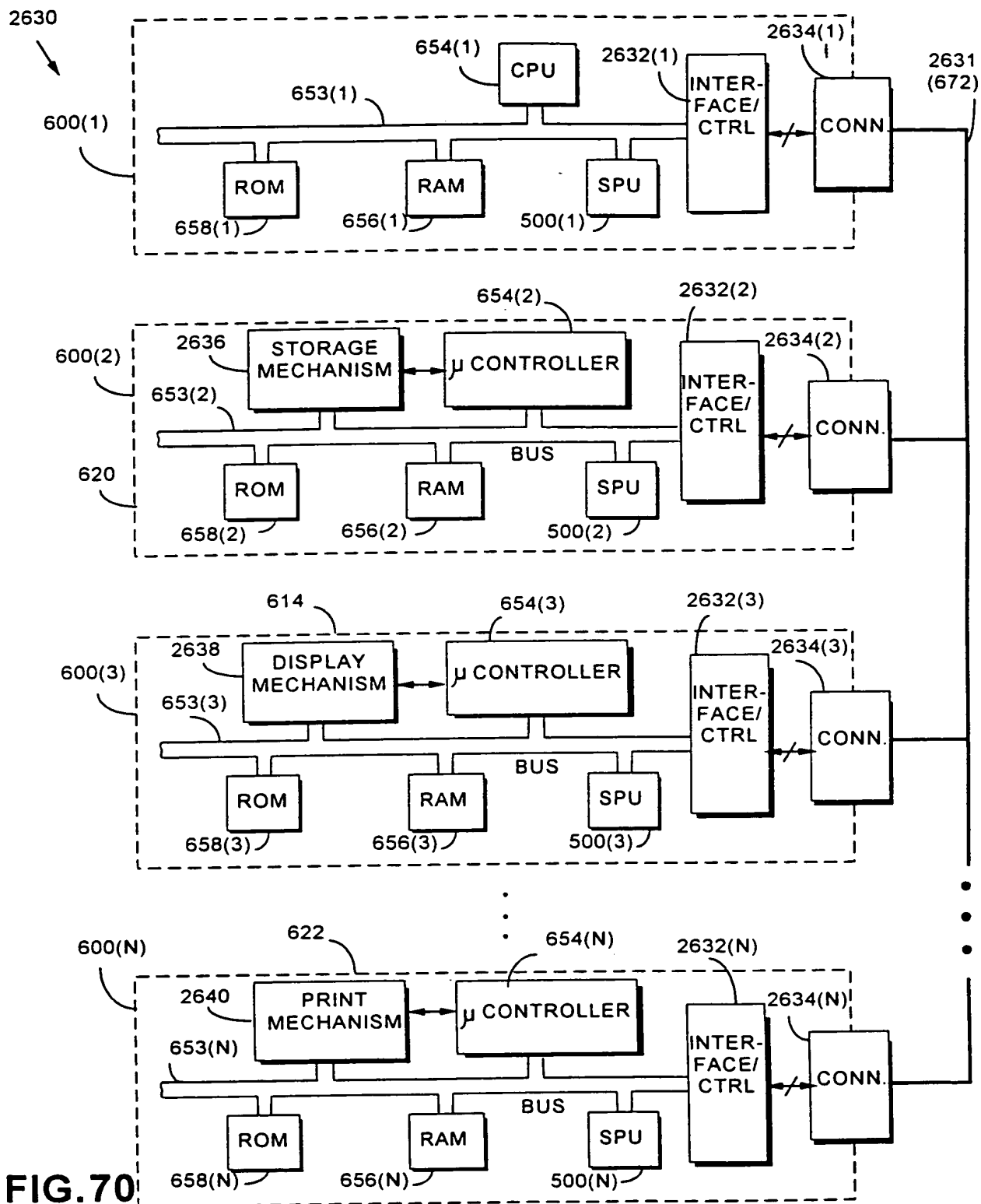
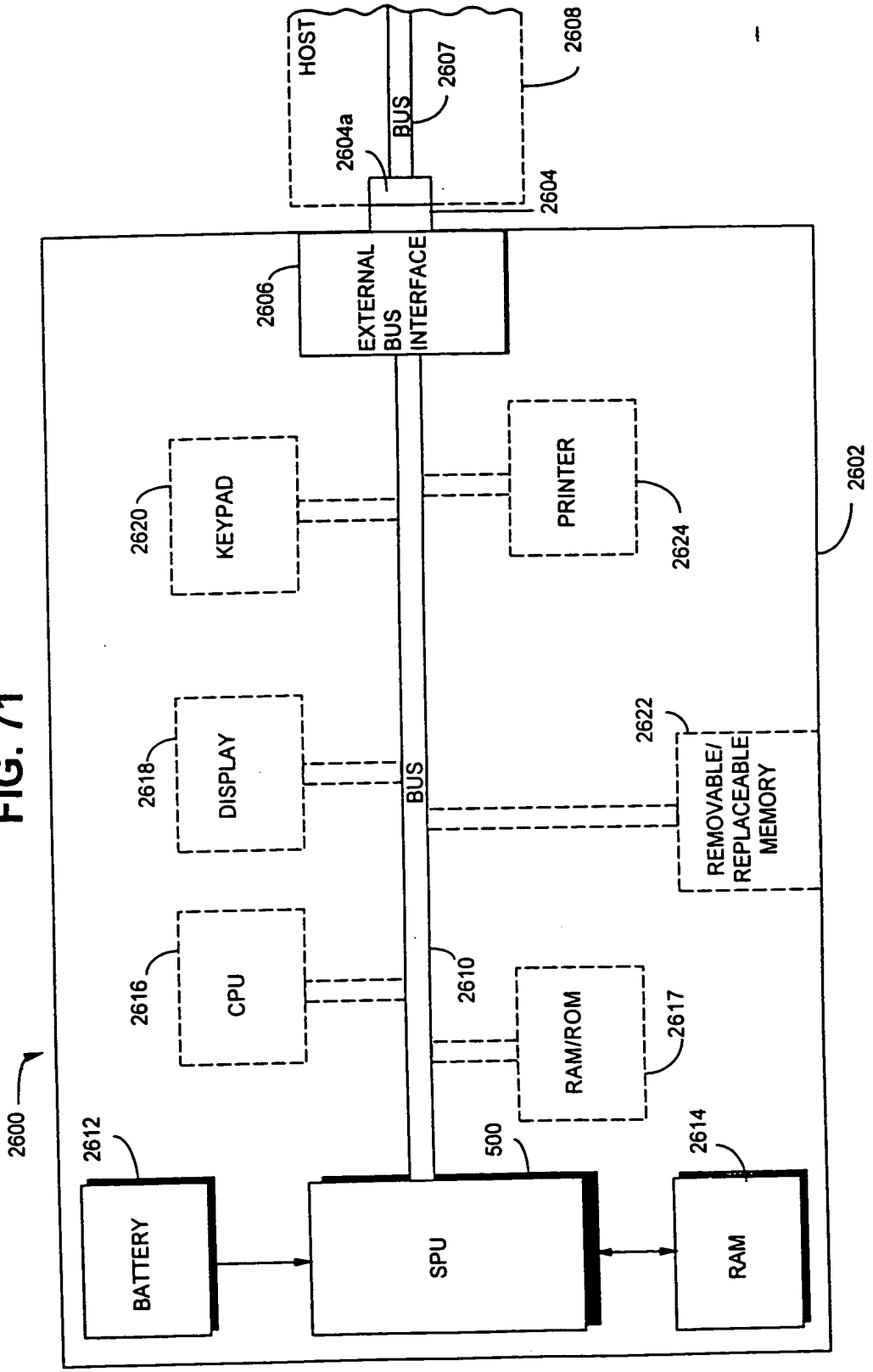


FIG. 70

FIG. 71



LOG IN USER INTERFACE

182

A login interface box with a thick border. It contains two input fields: 'USER NAME:' with the text 'SHEAR, V.' and 'PASSWORD:' with five asterisks. To the right of these fields are three buttons: 'LOGIN', 'CANCEL', and 'HELP'. At the bottom left is a checkbox labeled 'LOGIN AT STARTUP'.

FIG. 72A

FIG. 72B

A confirmation dialog box with a thick border. It features a warning icon (triangle with an exclamation mark) on the left. The main text reads 'YOU HAVE REQUESTED THESE PROPERTIES:' followed by 'LOONEY TUNES NEWS!'. Below this is a 'PROPERTY INFO' button. To the right are 'APPROVE' and 'SUSPEND' buttons. At the bottom, it says 'Your Cost: \$7.50' and 'MORE OPTIONS' with a question mark icon. Reference numerals 2660, 2662, and 2664 point to the 'CANCEL' button, the 'APPROVE' button, and the 'MORE OPTIONS' text respectively.

FIG. 72C

SET LIMITS:

SESSION DOLLAR LIMIT: \$	<div>2666</div> <div>50</div>	<div>2674</div> <div>OK</div>
TRANSACTION DOLLAR LIMIT: \$	<div>2668</div> <div>50</div>	<div>CANCEL</div>
TIME LIMIT (IN MINUTES):	<div>2670</div> <div>50</div>	
UNIT LIMIT:	<div>2672</div> <div>50</div>	<div>HELP!</div>

2666

2674

OK

CANCEL

HELP!

50

50

50

50

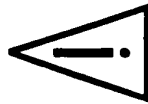
2668

2670

2672

05070552 40500

FIG. 72D



YOU HAVE REQUESTED THESE PROPERTIES:

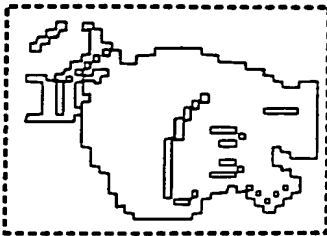
LOONEY TUNE NEWS!

PROPERTY INFO

YOUR COST : \$7.50

More Options ☒

Show Thumbnail



CANCEL

APPROVE

SUSPEND

PROPERTY:	SIZE:	PUBLISHER:	AMOUNT:	UNITS:	COST/UNIT:	TYPE:	USE?	LINKS:	HIST:
CHUCK JONES BIOGRA...	256KB	WARNER NEW MEDIA	64	KBYTE	\$1.25	PREVIEW	✓		●
▼ BUGS BUNNY.JPE...	1MB	WARNER NEW MEDIA	1	RECORD	\$5.00	DISPLAY	✓	▲	●
BUGS BUNNY.JPEG...	1MB	WARNER NEW MEDIA	10	RECORD	\$3.50	DISPLAY		▲	●
BUGS BUNNY.JPEG...	1MB	WARNER NEW MEDIA	25	RECORD	\$2.50	DISPLAY		▲	●
FRIZ FRELENG BIOGRA...	256KB	WARNER NEW MEDIA	120	SECTOR	\$5.00	PRINT			
TEX AVERY BIOGRAP...	256KB	WARNER NEW MEDIA	50	PERCENT	\$2.50	COPY		▲	
► DUCKI RABBITI DU...	64MB	WARNER NEW MEDIA	7.0	MINUTE	\$7.50	COPY-PRO			
MEL BLANC BIOGRAPH...	256KB	WARNER NEW MEDIA	1	SPECIAL	\$25.25	INSTALL		▲	
LOONEY TUNES DATAB...	600MB	WARNER NEW MEDIA	1	OBJECT	\$2000.00	ALL		▲	●

SET LIMITS...	SHOW BUDGETS	ACQUIRE BUDGET...	HISTORY...	TRANSFER...	PREFERENCES...	FEEDBACK...	HELP!
---------------	--------------	-------------------	------------	-------------	----------------	-------------	-------

FIG. 73

The diagram illustrates a data structure 3000, which is a vertical stack of blocks. The structure is organized as follows:

- PUBLIC HEADER** (802)
- PRIVATE HEADER** (804)
- PRIVATE BODY** (806)

The Private Body is further divided into several sections:

- Rules for Container 300** (806a), **Rules for Rights Negotiation** (806b), and **Rules for Container 300w** (806c)
- Rules for Container 300x** (806d), **Rules for Container 300y** (806e), and **Rules for Container 300z** (806f)
- CONTENT OBJECT** (300z), **SOFTWARE AGENT** (300z(2)), and **RULES FOR AGENT EXECUTION** (300z(1))
- CONTENT OBJECT** (300y), **INFORMATION (ROUTING) LOCATIONS AND RELATED DATA** (300y(2)), and **RULES FOR INFORMATION SEARCH** (300y(1))
- CONTENT OBJECT** (300x), **INFORMATION RETRIEVED** (300x(2)), and **RULES FOR INFORMATION RETRIEVED** (300x(1))
- ADMIN. OBJECT** (300w), **AUDIT HISTORY OF AGENT EXECUTION** (300w(2)), and **RULES FOR AUDIT RETURNED** (300w(1))

300

806a

806b

806d

806e

300z

300y

300x

300w

FIG. 74

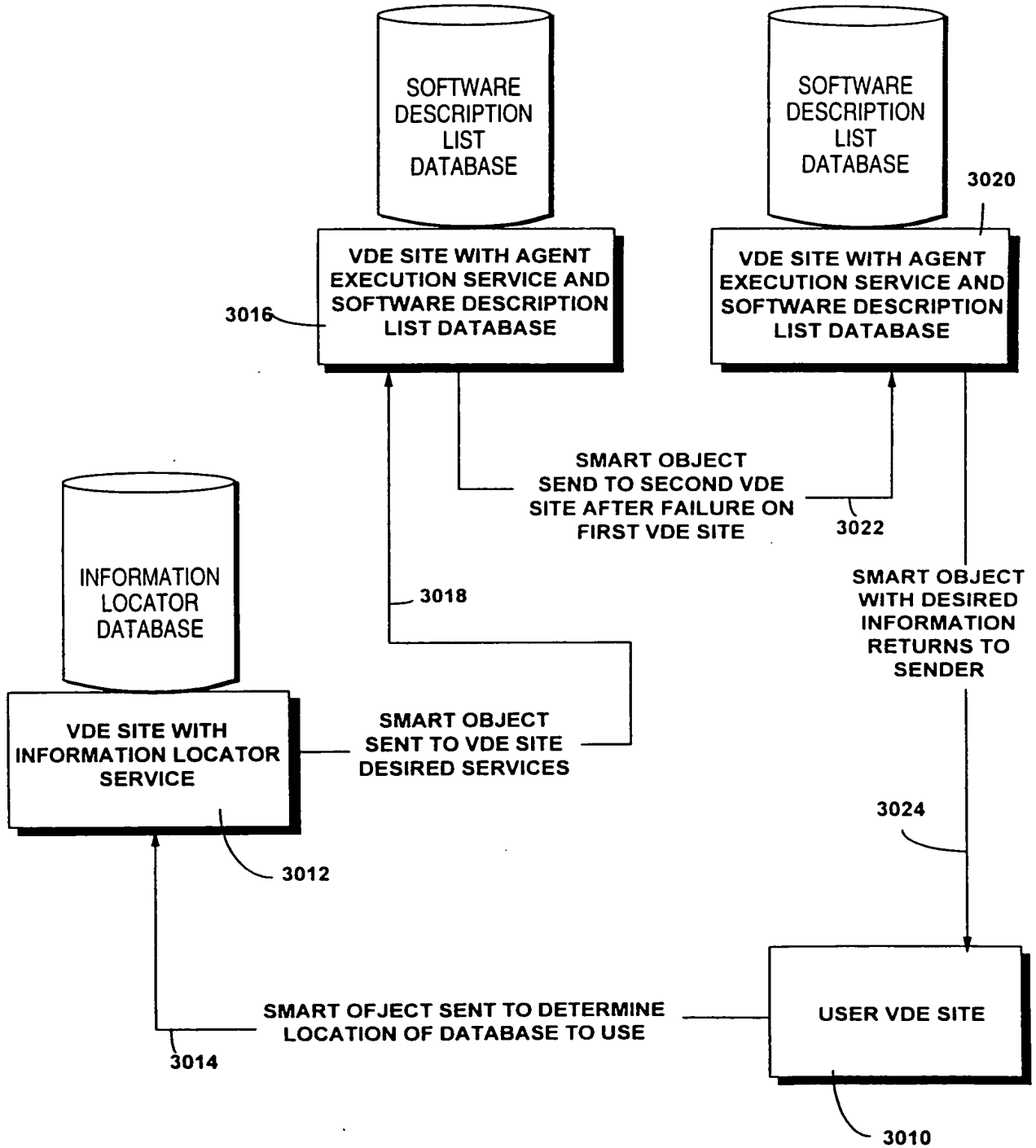


FIG. 75A

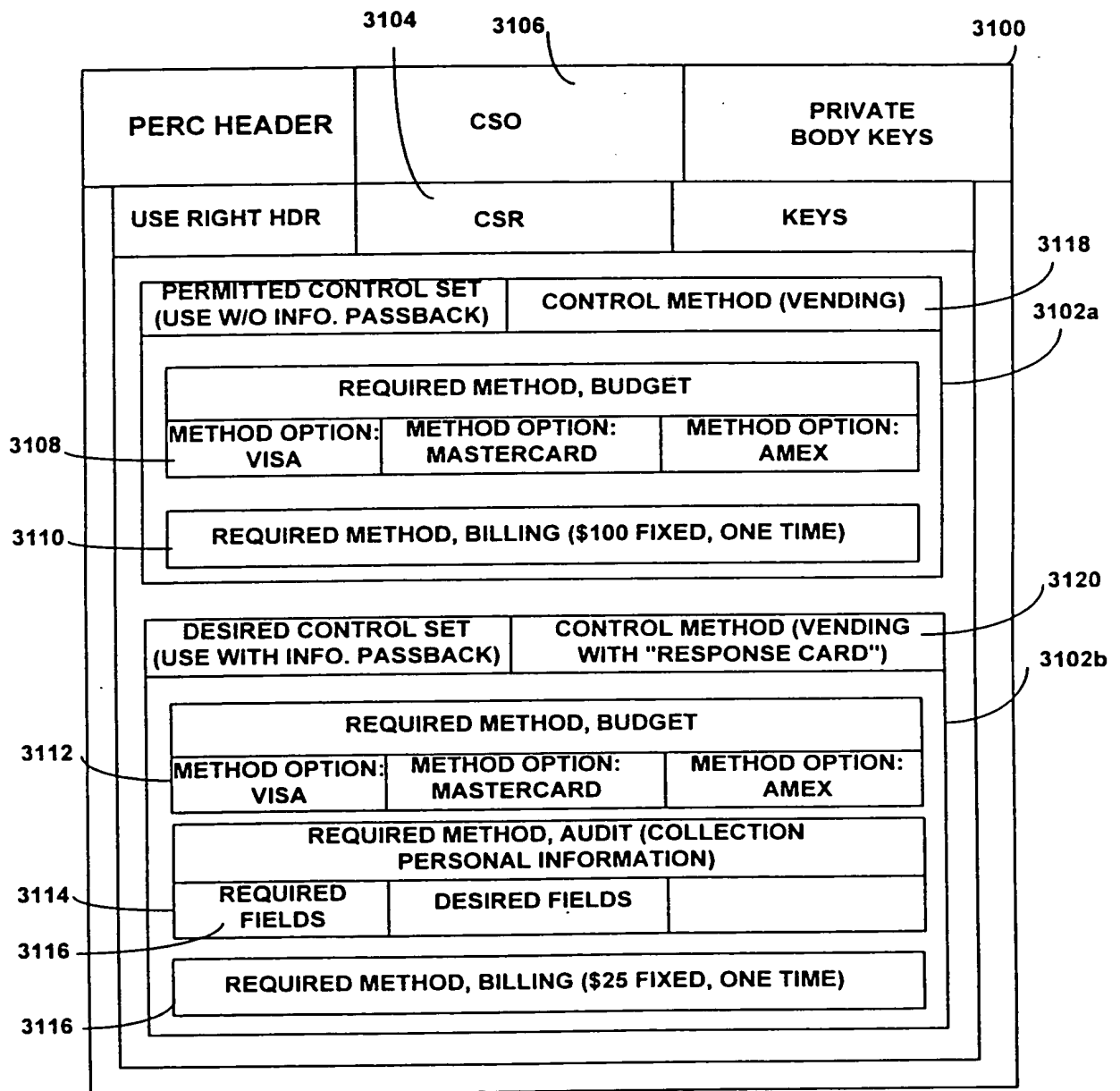


FIG. 75B

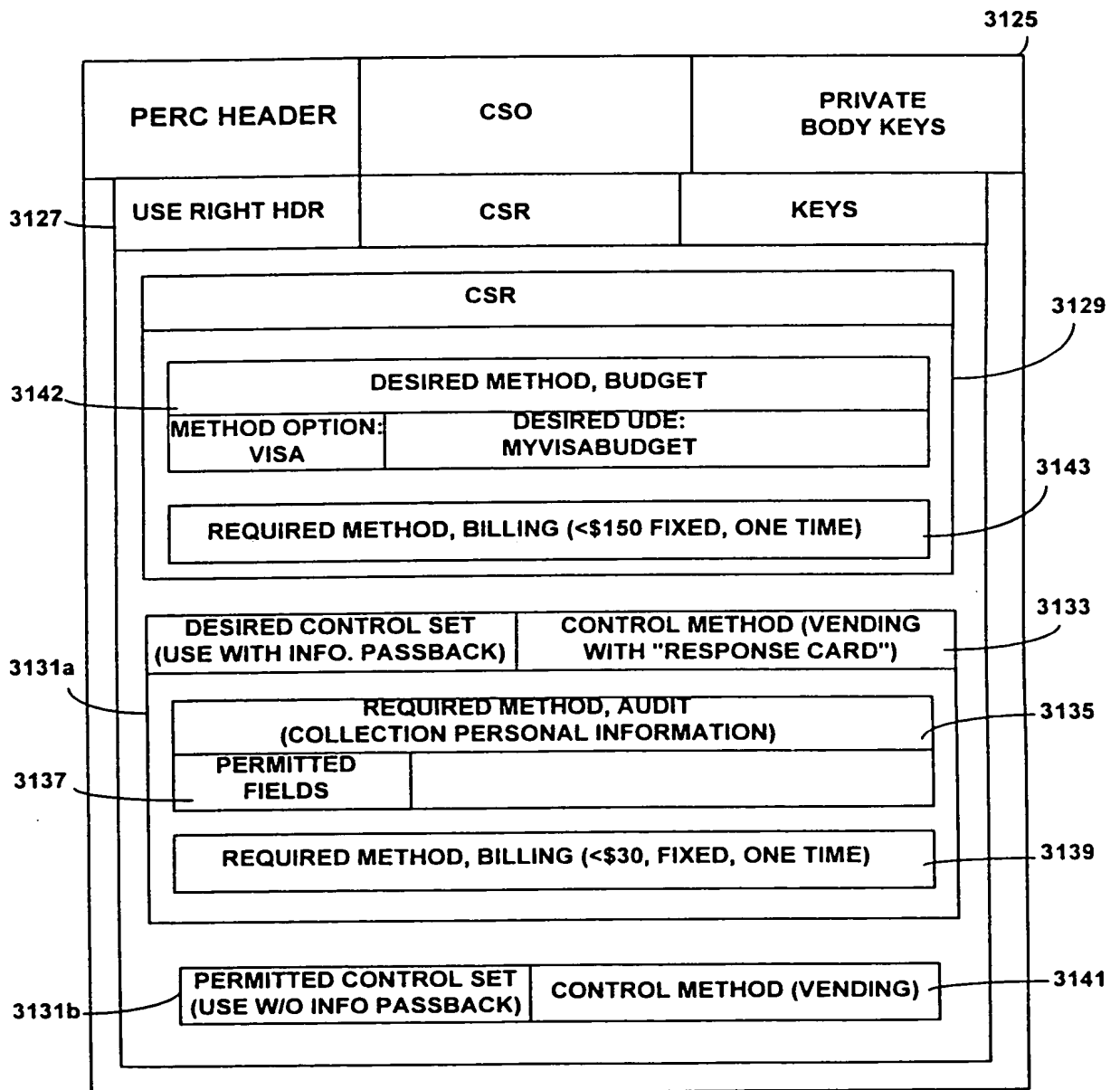


FIG. 75C

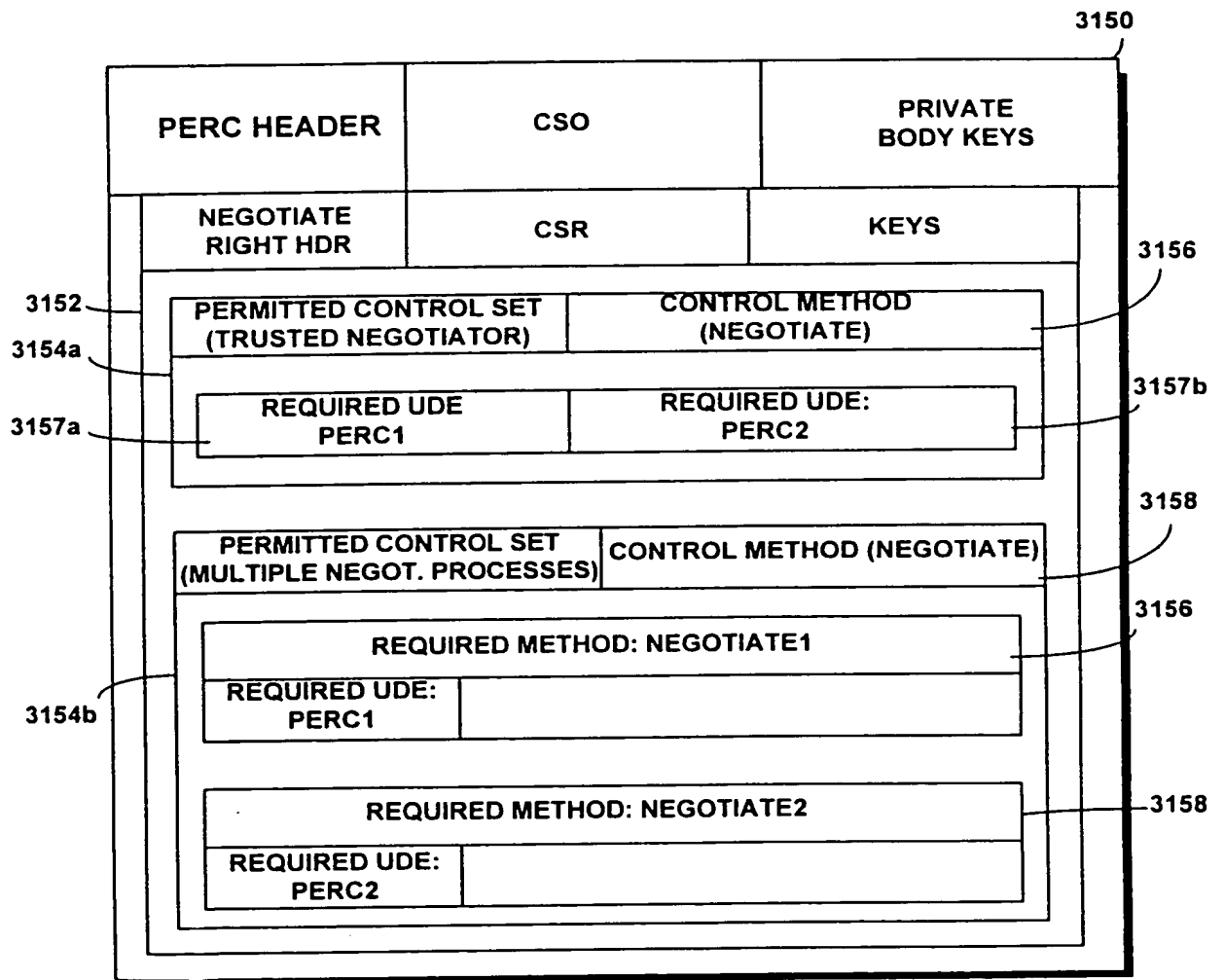


FIG. 75D

URT HEADER		CSO		DIGITAL SIGNATURE		3160
USE RIGHT HDR		CSR				
CONTROL SET(USE WITH INFO. PASSBACK)		CONTROL METHOD(VENDING WITH "RESPONSE CARD")				
REQUIRED METHOD, BUDGET						
METHOD OPTION: VISA		DESIRED UDE: MYVISABUDGET				
REQUIRED METHOD, AUDIT (COLLECTION PERSONAL INFORMATION)						
PERMITTED FIELDS						
REQUIRED METHOD, BILLING(\$25, FIXED, ONE TIME)						

3162

3164

3166

3170

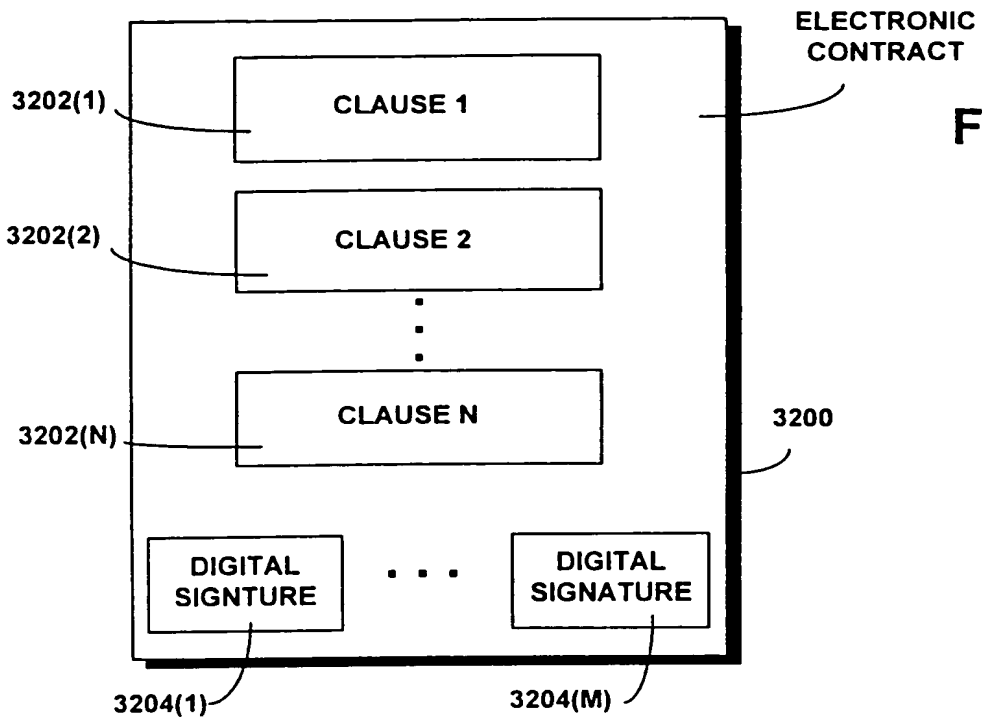


FIG. 75E

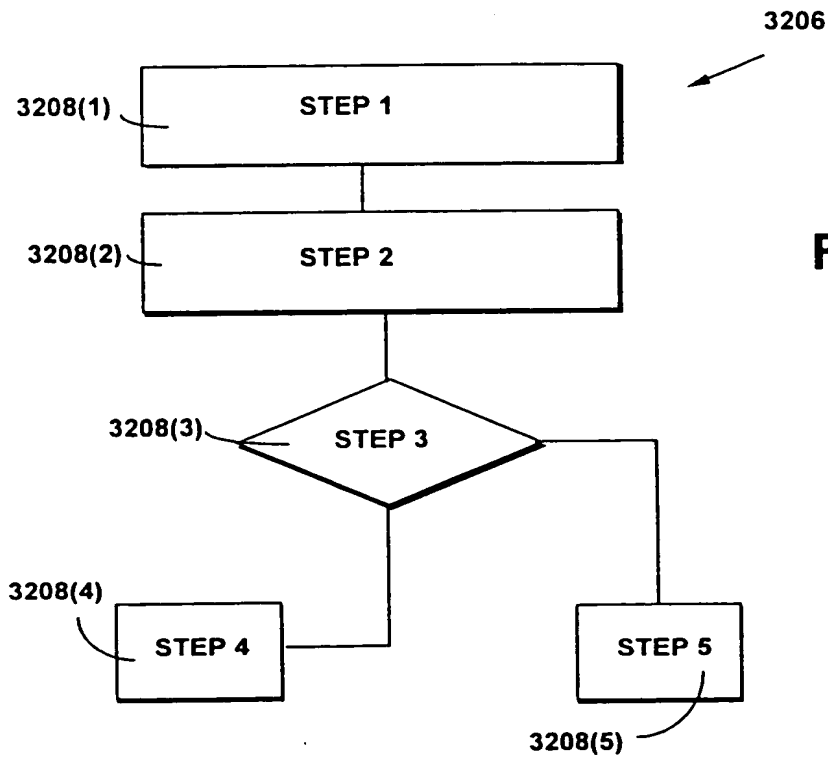


FIG. 75F

FIG. 76A

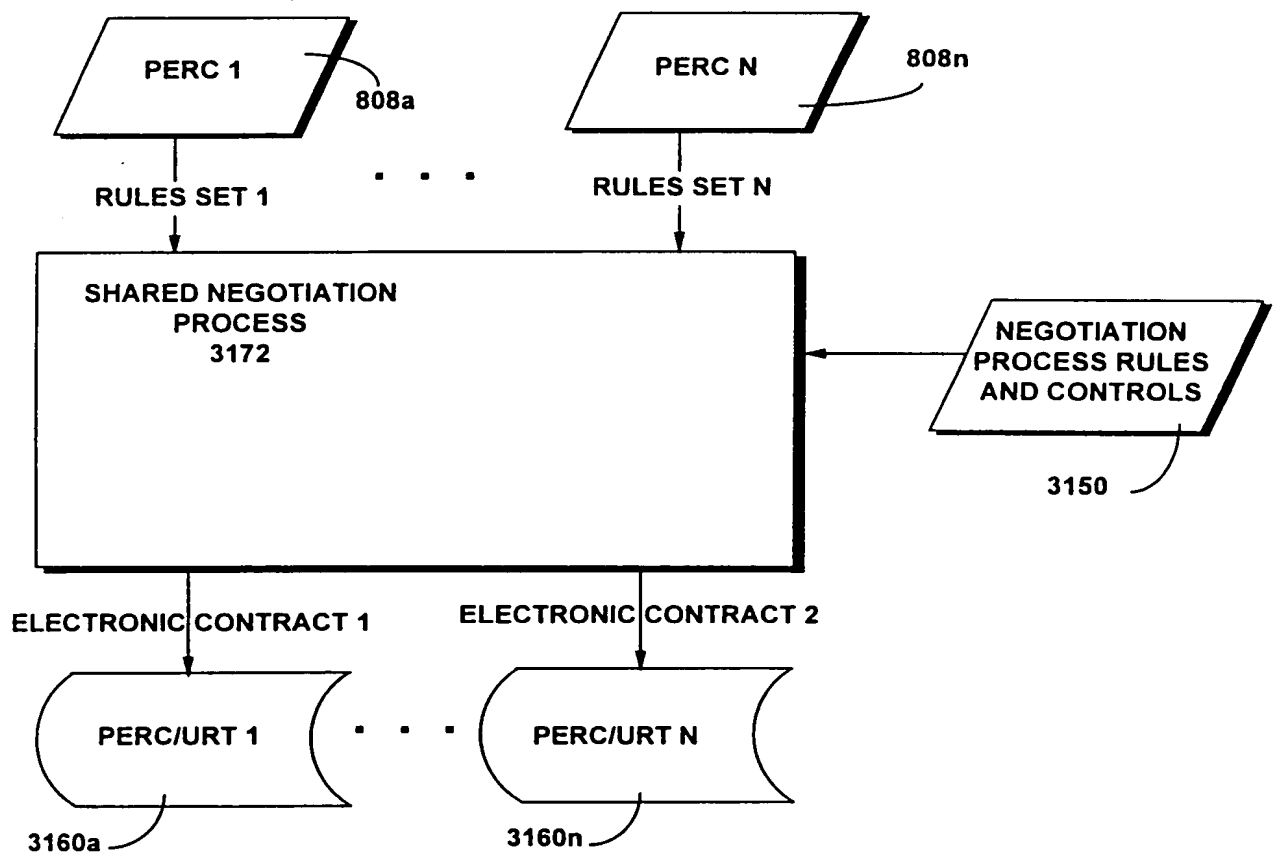
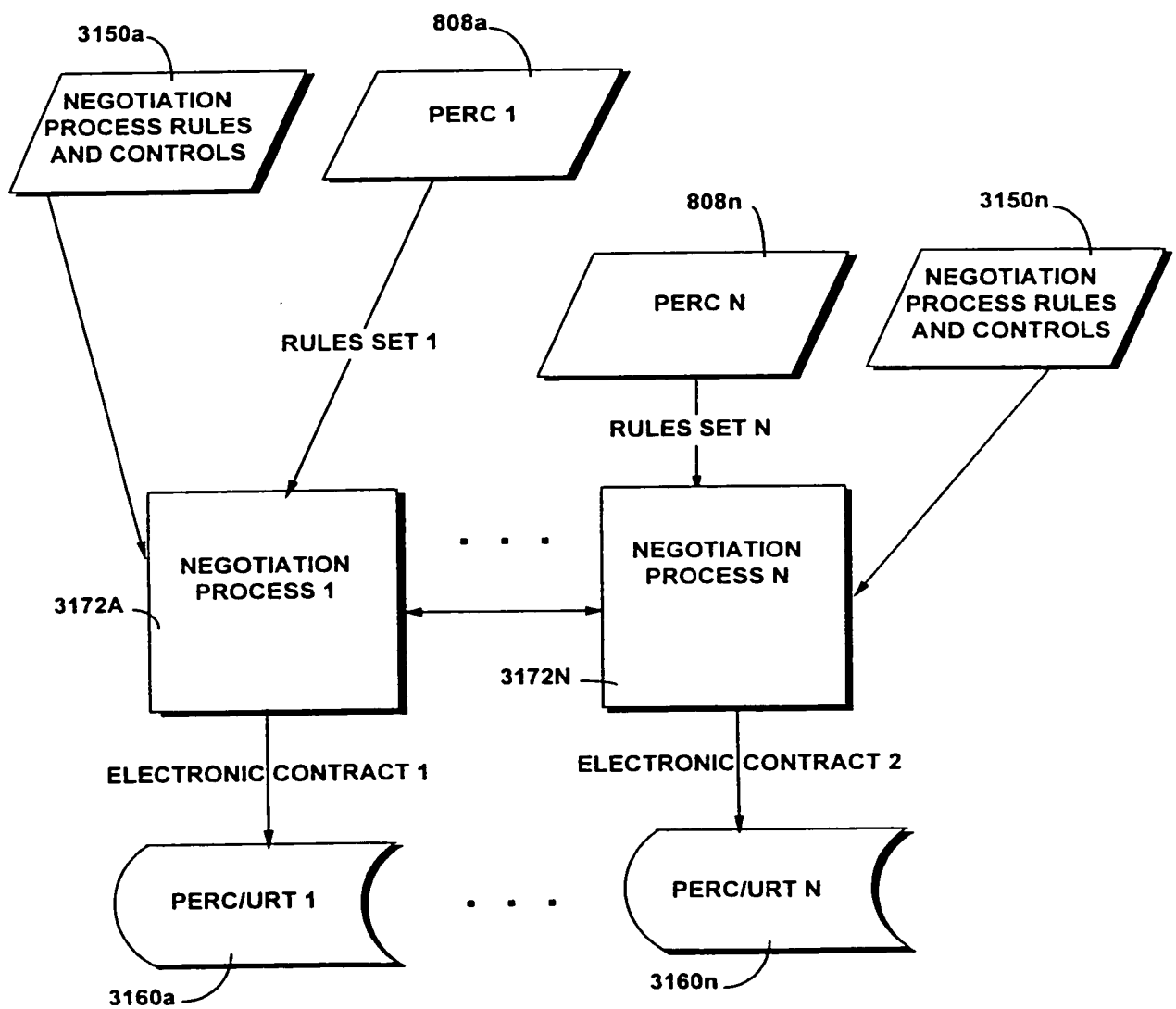
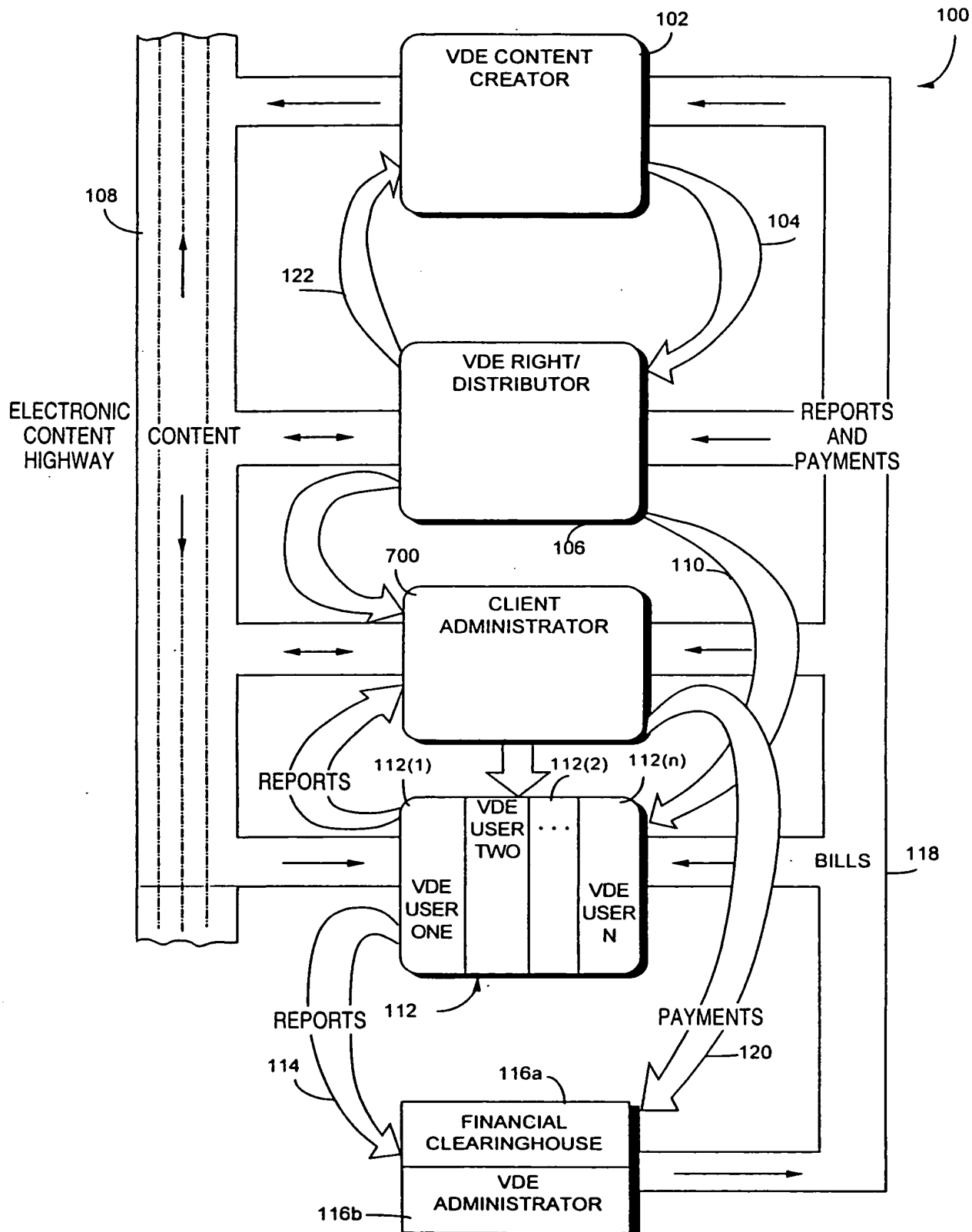


FIG. 76B



of which the first is the most important, and the second is the most important of the first.



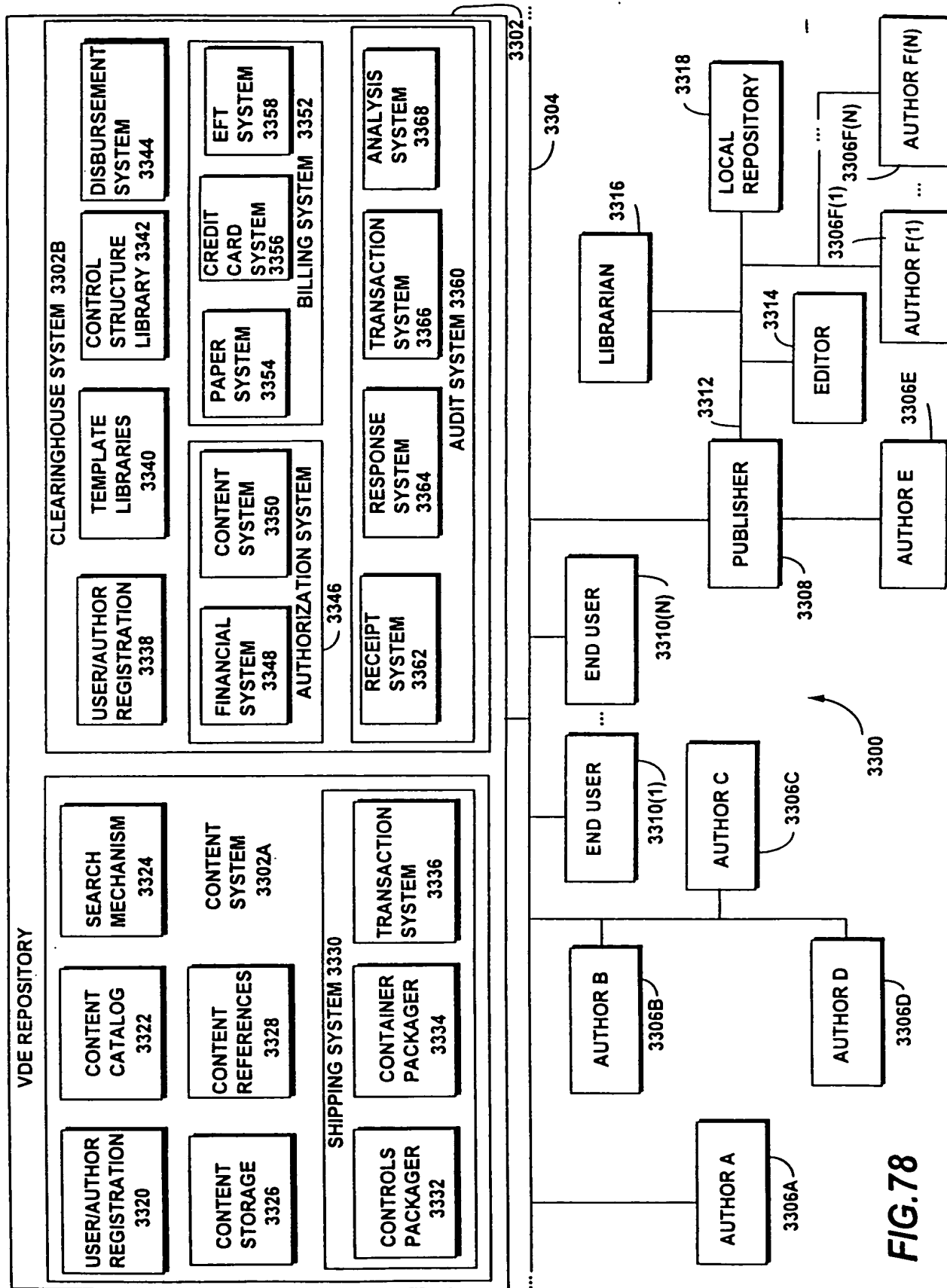
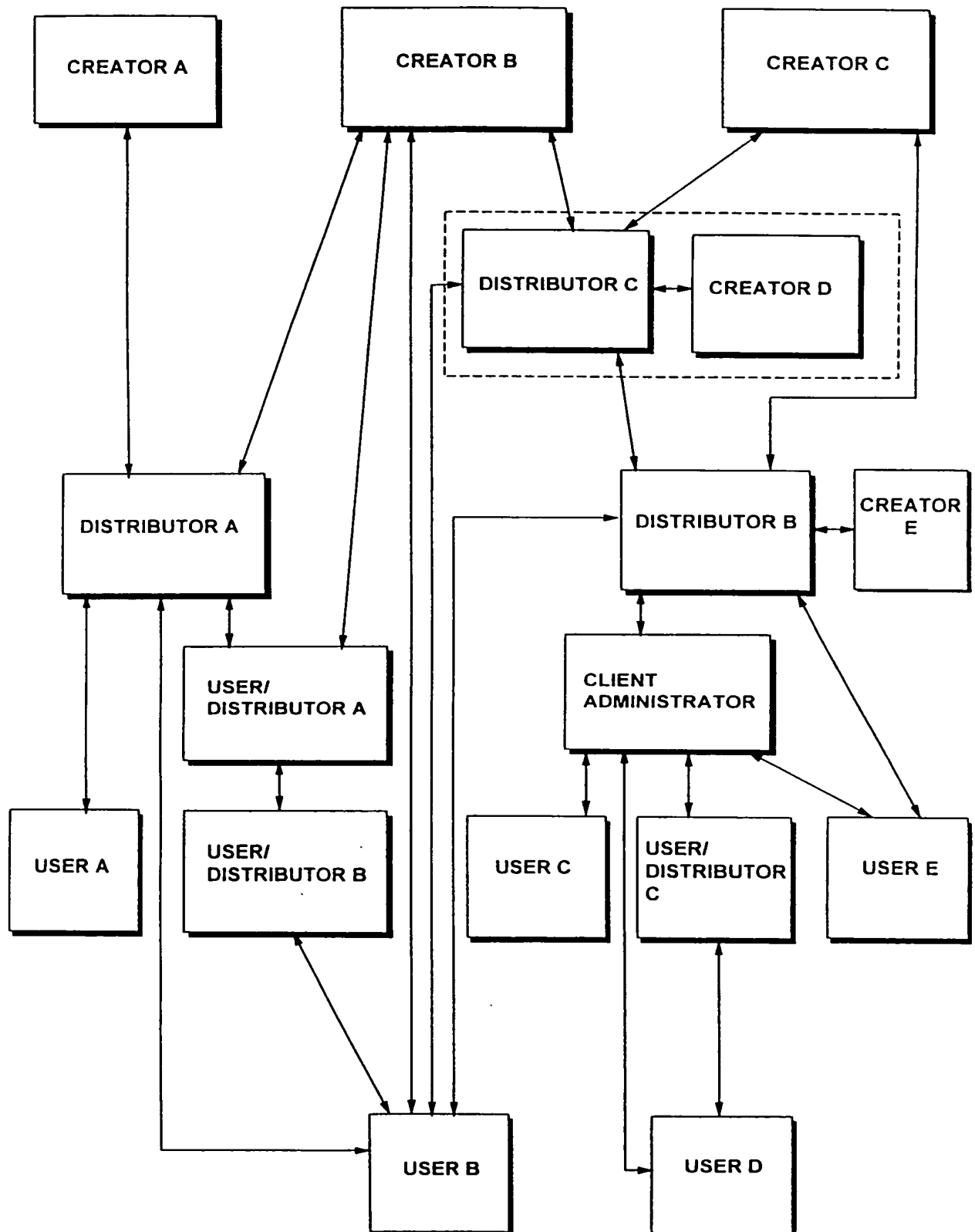


FIG.78

FIG. 79



US 7,625,103 B2

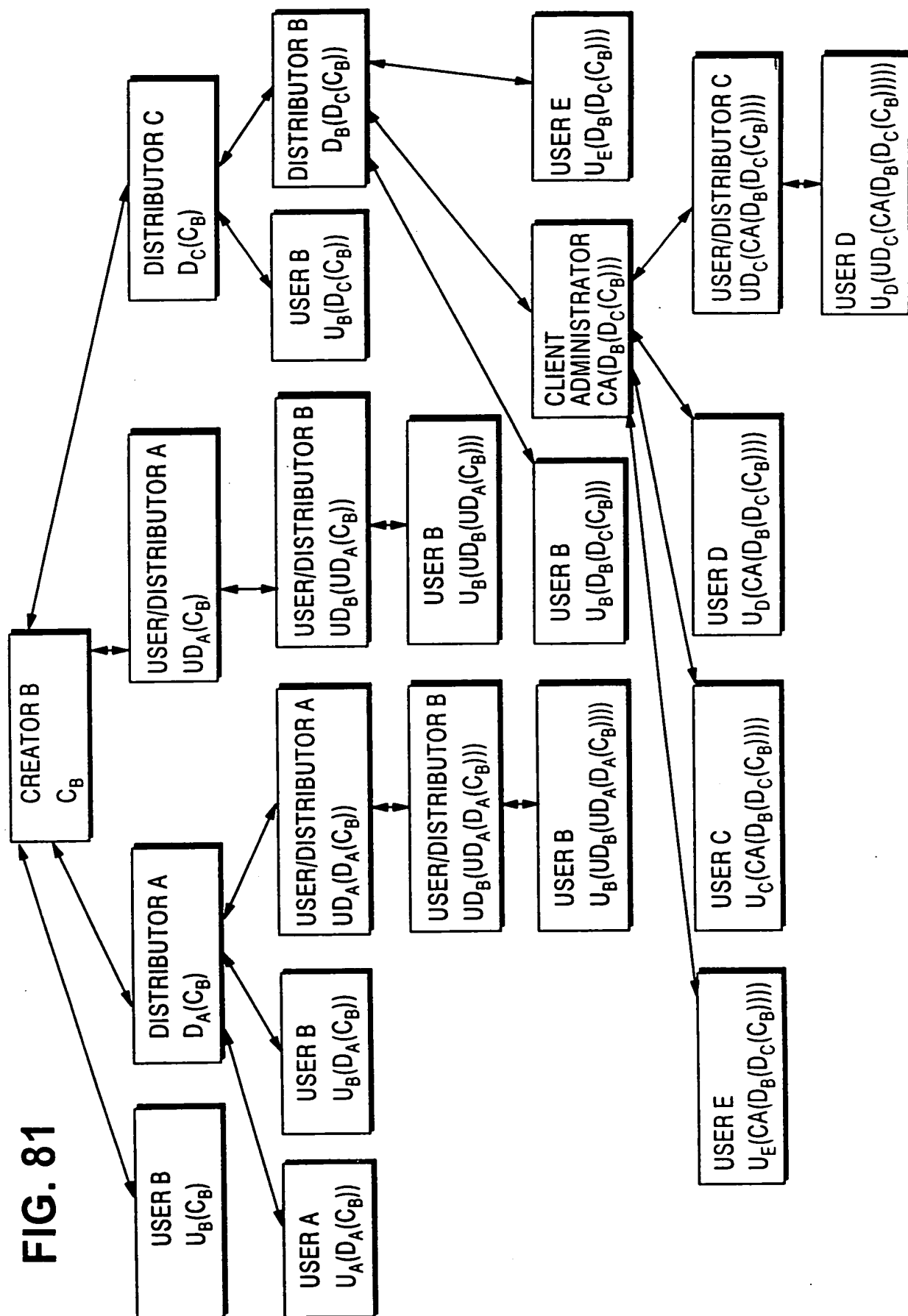


FIG. 82

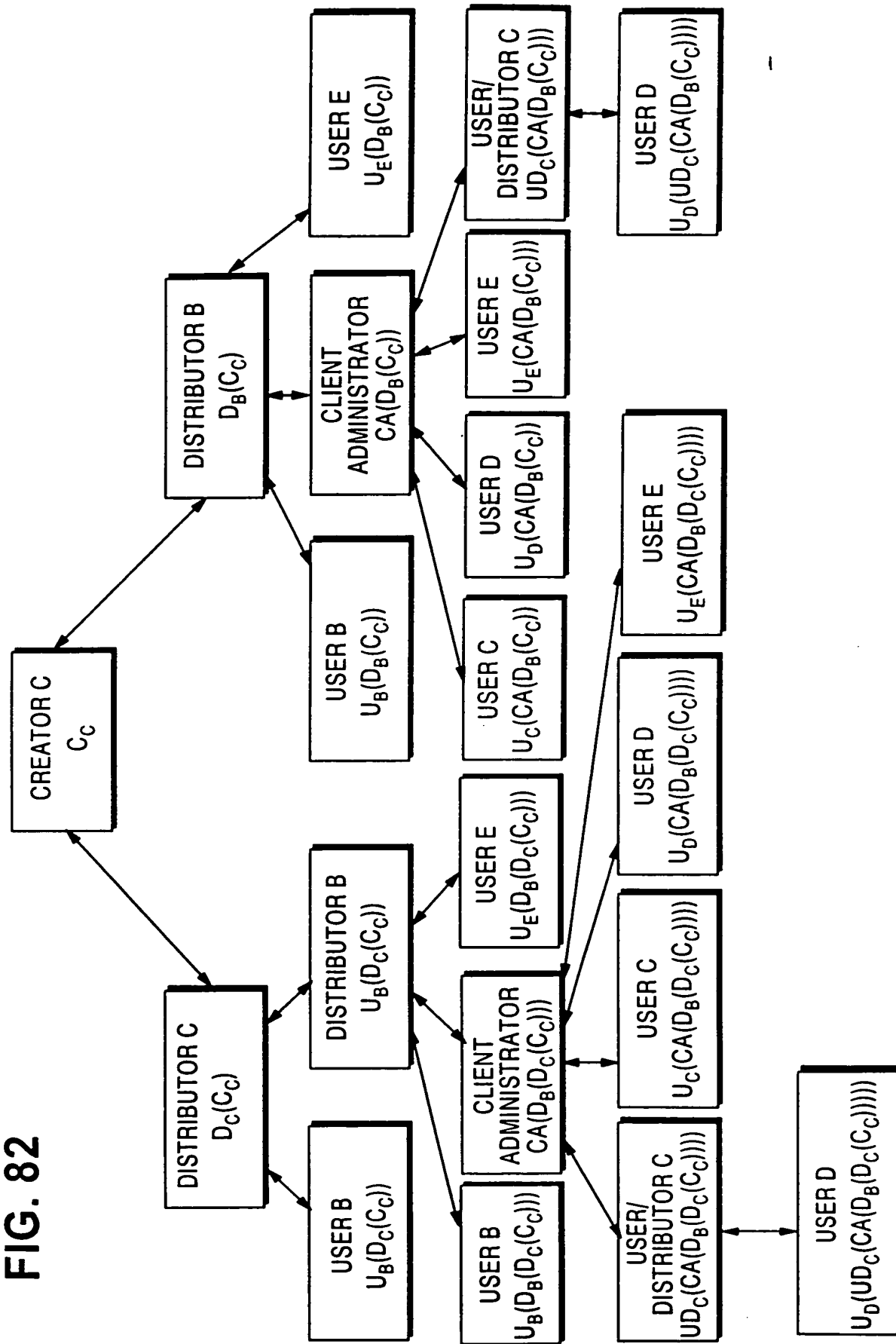
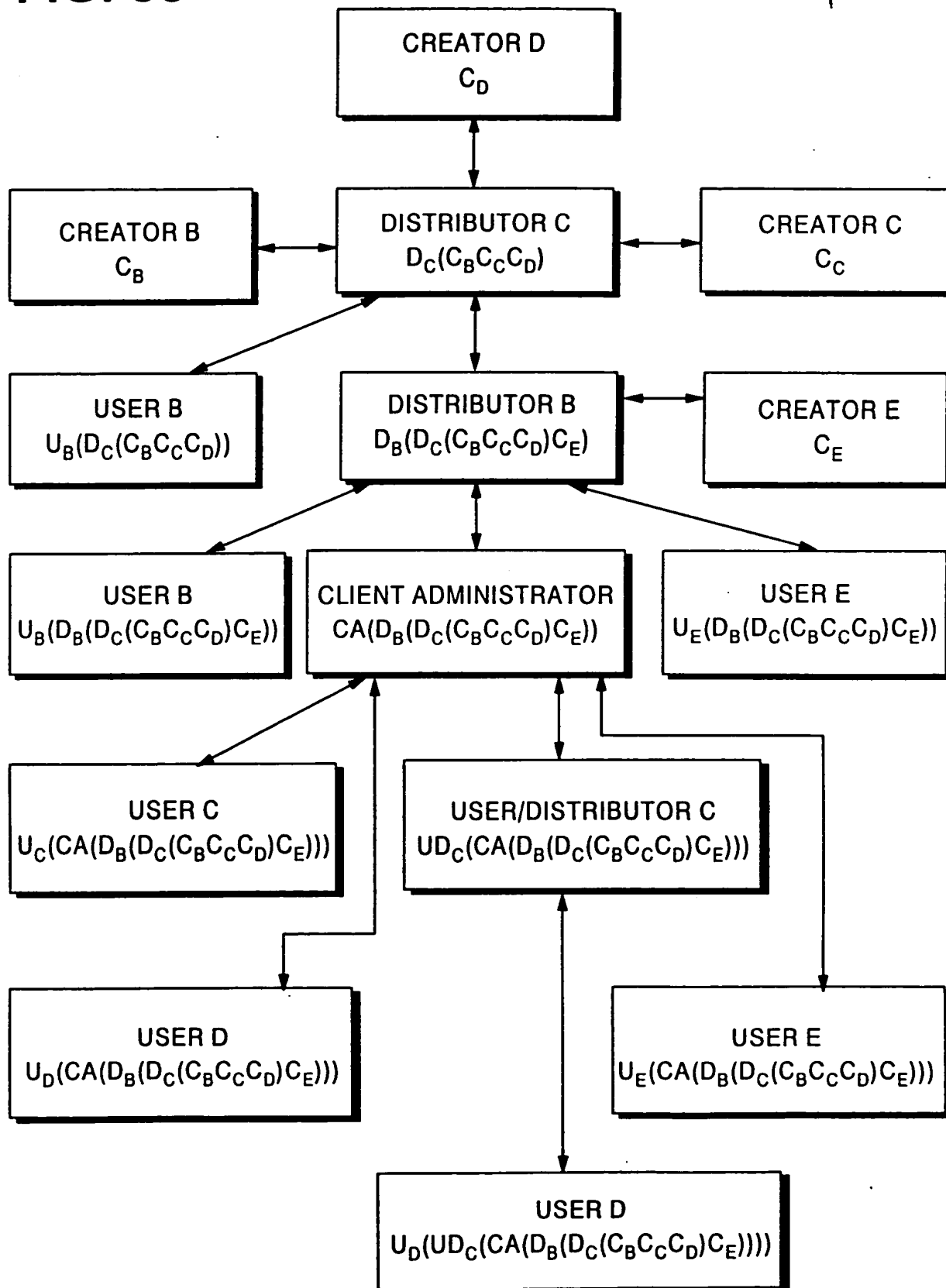
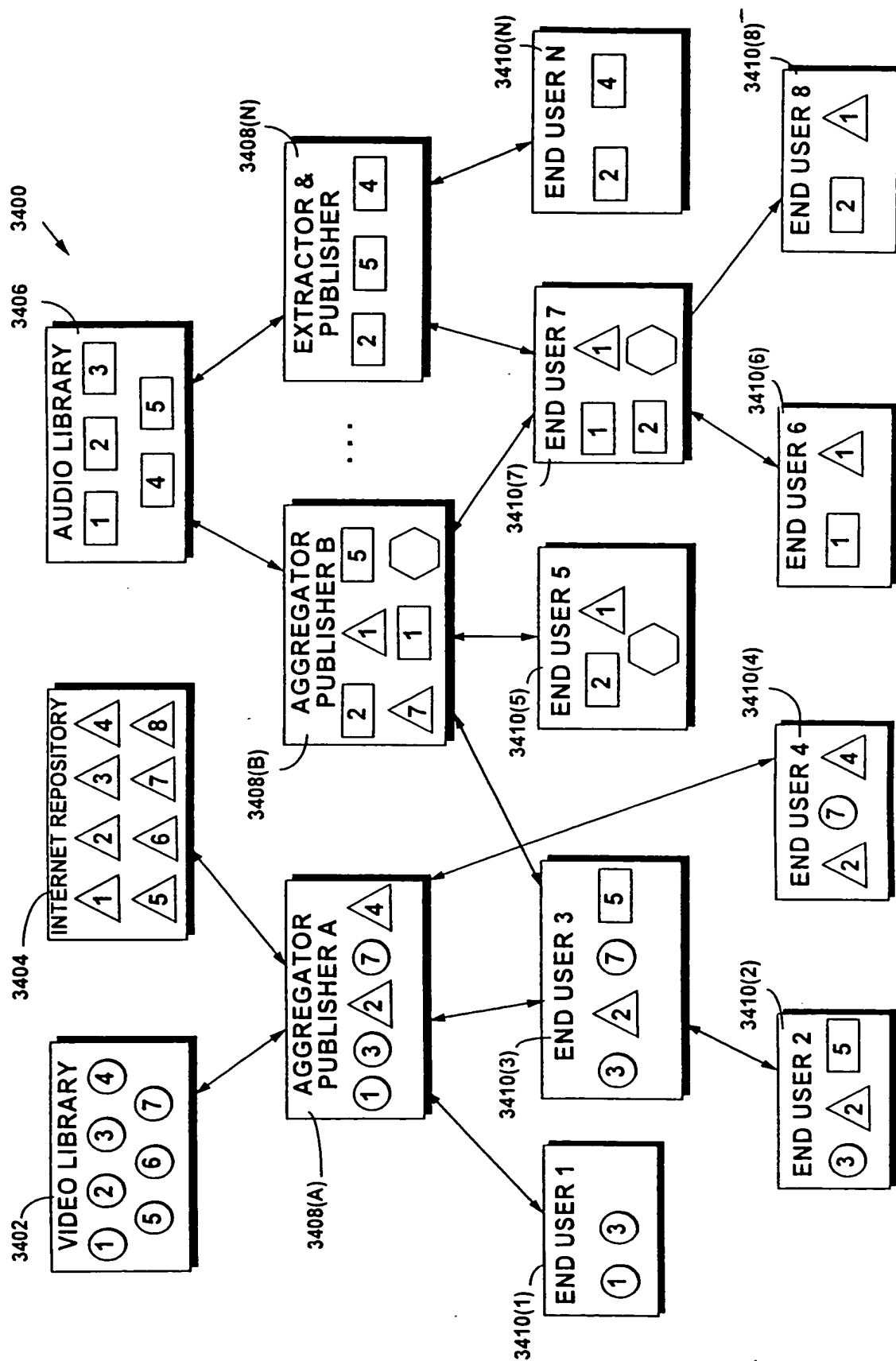


FIG. 83



02676222 100377

FIG. 84



1. The first step is to identify the problem. This involves understanding the situation and the goals that need to be achieved.

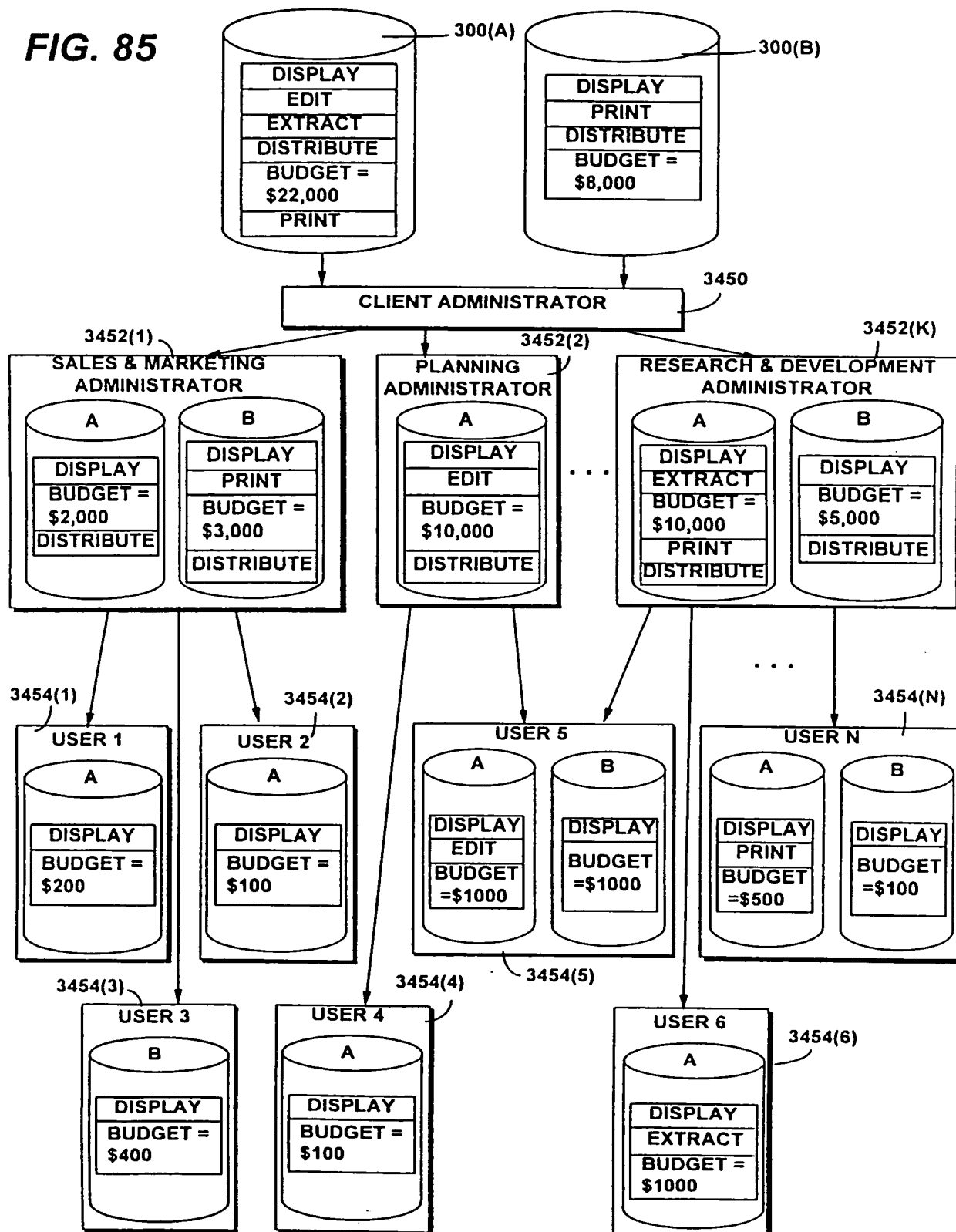


FIG. 86

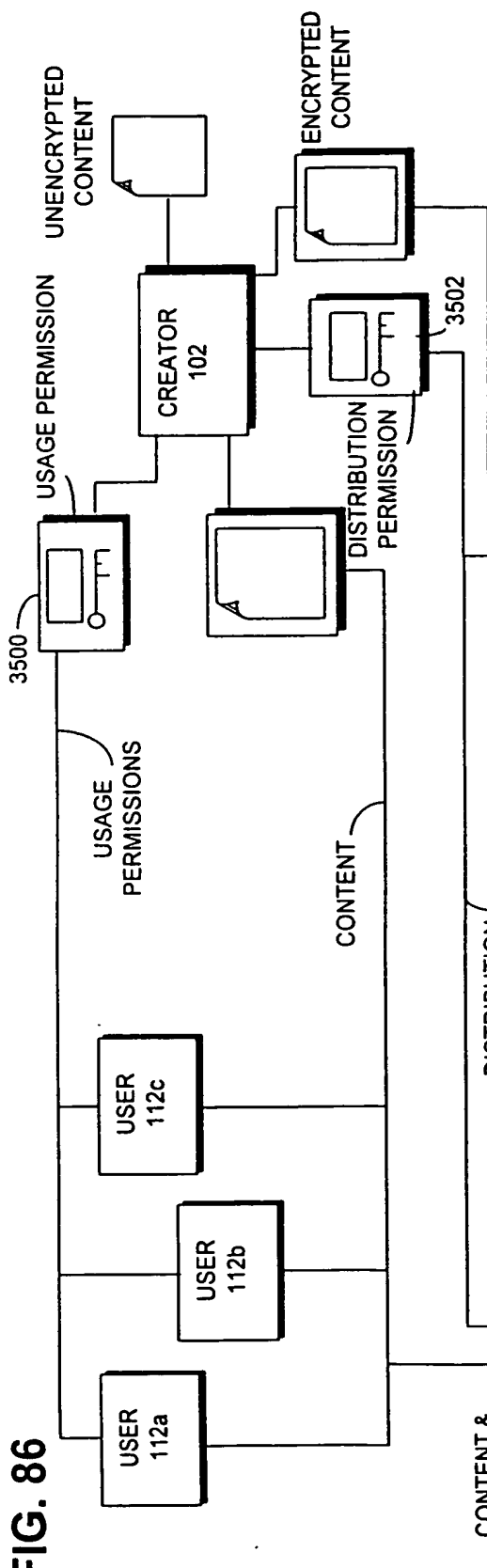


FIG. 86A

